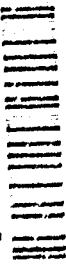


AD-A249 095



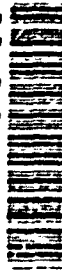
14th International Conference on the Numerical Simulation of Plasmas

Loews Annapolis Hotel
Annapolis, Maryland, USA

September 3-6, 1991

This document has been approved
for public release and sale; its
distribution is unlimited.

92-03176



92 2 02 022

PROCEEDINGS

14th International Conference on the Numerical Simulation of Plasmas

**Loews Annapolis Hotel
Annapolis, Maryland, USA
September 3-6, 1991**

A Meeting of

**The American Physical Society
Topical Group on Computational Physics**

Sponsored by

**Science Applications International Corporation
Naval Research Laboratory
Office of Naval Research**

Accession For	
NTIS	CRA&I
DTIC	TAB
Unannounced	<input type="checkbox"/>
Justification	<input type="checkbox"/>
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	



Statement A per telecon
Dr. Charles Roberson ONR/Code 1112
Arlington, VA 22217-5000
NMW 4/16/92

Program Committee

A. Adolf, CEA-CELV
M. Ashour-Abdalla, UCLA
J. Denavit, LLNL
R. Faehl, LANL
I. Haber, NRL
W. Herrmannsfeldt, SLAC
J. Johnson, PPPL
A. Mankofsky, SAIC
D. Seidel, SNL

Conference Staff

A. Mahaffey, SAIC
J. Staudinger, SAIC
B. Pitcher, NRL
R. Bain, NRL

Organizing Committee

A. Mankofsky, SAIC
I. Haber, NRL

GENERAL INFORMATION

Welcome to the 14th International Conference on the Numerical Simulation of Plasmas. The conference convenes at 7:30pm on Tuesday, September 3, 1991 at Loews Annapolis Hotel in Annapolis, Maryland, and adjourns at 1:00pm on Friday, September 6, 1991. Below are some general facts for your information:

Registration: The conference registration desk is located in the Atrium, between Regatta Ballroom C and the Windjammer. Registration packets are ready for all those who have preregistered; on-site registration is \$160. As the conference is self-supporting, we ask that everyone please remember to register. Also available at the registration desk are local sightseeing information, restaurant guides, and travel information. The registration desk is open throughout the conference.

Conference Organization: All technical sessions will be held in the Regatta Ballroom complex: oral presentations and panel discussion in ballrooms A and B, and posters in ballroom C. Please refer to the accompanying hotel floor plan for room locations. The program begins each morning at 8:30am, following a continental breakfast (included in the

registration fee) available in the Atrium. Coffee, tea, and soft drinks will be provided during the sessions. Note that late Wednesday afternoon is free for exploring Annapolis; a poster session is scheduled for later that same evening.

Program: A total of 85 presentations is scheduled. This includes 2 special invited talks, 24 invited oral papers, 58 posters, and a panel discussion on UNICOS and distributed computing. Since we were able to accommodate all those who requested oral presentations, the schedule is fairly tight; we therefore request your help in keeping orals within the 25-minute time limit, which includes 5 minutes for questions. Poster sessions are 2 hours in length, while the panel discussion is 1 hour 30 minutes long. There are no parallel sessions.

Social Events: A registration reception will be held in the Windjammer on Tuesday evening for arriving participants and guests. The conference banquet (also included in the registration fee) is scheduled for Thursday evening in Regatta Ballrooms A and B. Additional banquet tickets (\$25 each) may be purchased at the registration desk until 10:00am on Wednesday.

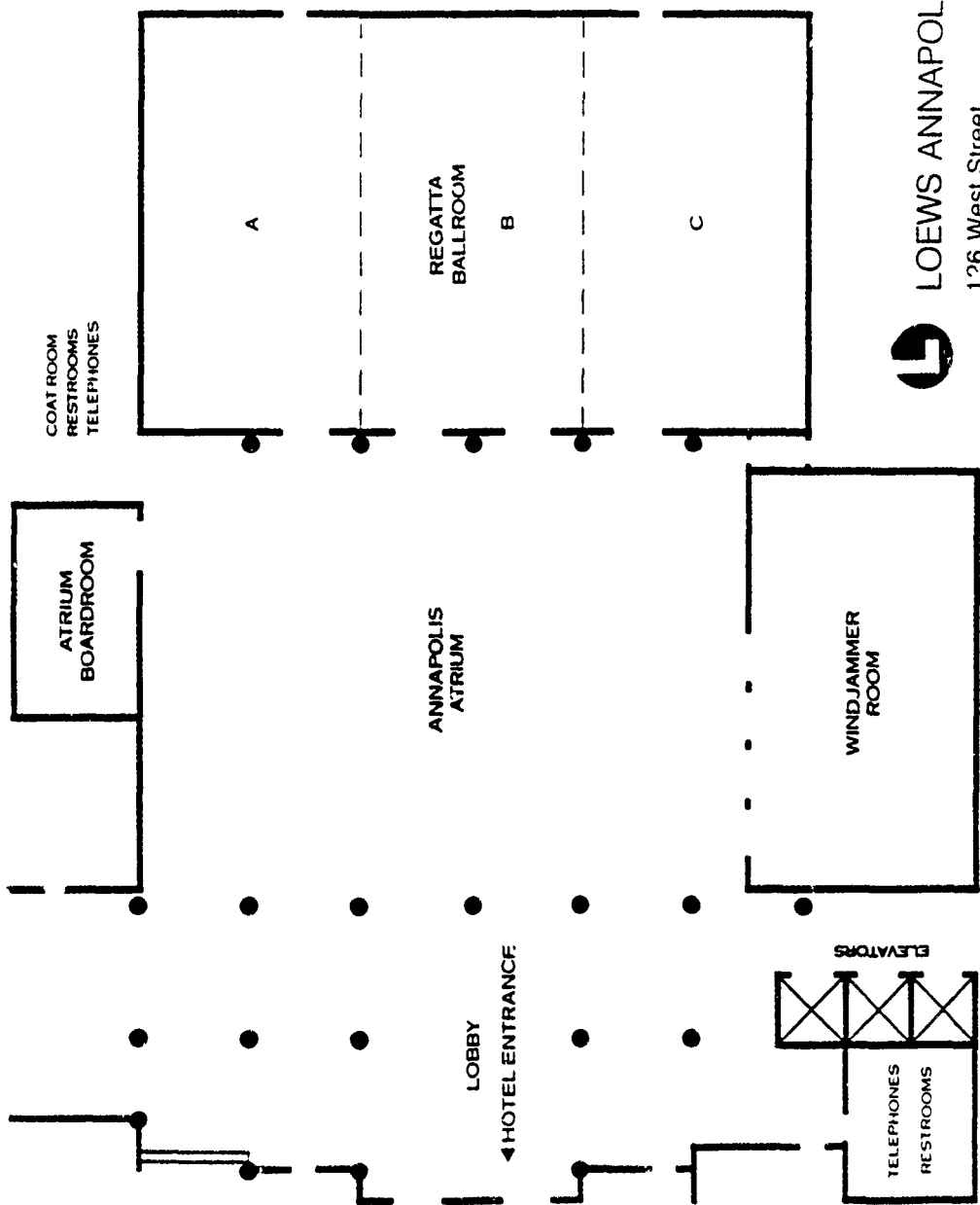
Business Meeting: The business meeting will be held on Thursday at 5:30pm in the Catamaran Room. The scheduling, location, and format of the next conference will be chosen at this meeting. All are invited.

Facilities: An overhead projector, a 35mm slide projector, VHS video equipment, and monitors will be available throughout the conference. Those who requested special equipment have been contacted individually regarding arrangements. A copying service (Minuteman Press) is located a block from the hotel.

Messages: A message board is located in the conference registration area. The telephone number of the hotel is (301) 263-7777 -- ask for the plasma simulation conference registration desk. The hotel fax number is (301) 263-0084.

Acknowledgements: We would like to express our gratitude to the many people who helped with the organization and logistics of this conference. In particular, we thank Anita Mahaffey, Judy Staudinger, Bertha Pitcher, Ruth Bain, Priscilla Wallace, and Twilla Hepler, who endured endless crises and are just now emerging from beneath mountains of paperwork (not a minute too soon for Anita, who is nearly 8 months pregnant!). It was a pleasure working with everyone on the program committee, whose

input was invaluable and who succeeded in arranging an excellent technical program. We also thank the staff of Loews Annapolis Hotel for their assistance. Finally, we are grateful to the SAIC Applied Physics Operation, the NRL Plasma Physics Division, and the Office of Naval Research for their support.



LOEWS ANNAPOLIS HOTEL

126 West Street
Annapolis, Maryland 21401
Telephone (301) 263-7777
FAX (301) 263-0084

14th International Conference on the Numerical Simulation of Plasmas

Loews Annapolis Hotel
Annapolis, Maryland, USA
September 3-6, 1991

Tuesday, September 3, 1991, 7:30pm - 9:30pm

Registration -- Atrium

Reception -- Windjammer

Continental Breakfast in the Atrium each morning at 8:00am

Wednesday, September 4, 1991, 8:30am - 12:40pm

Regatta Ballrooms A & B

8:30 Opening Remarks -- A. Mankofsky

Special Invited Talk

8:40 Numerical Simulation of Semiconductor Devices -- Jeffrey Frey,
Univ. of Maryland

Session OW

Invited Talks -- J. Denavit, presiding

9:25 OW1 Numerical Simulations of the Vlasov-Maxwell Equations
by Coupled Particle-Finite Element Methods on
Unstructured Meshes -- P. Degond

9:50 OW2 Electromagnetic PIC Simulations Using Finite Elements on
Unstructured Grids -- J. Ambrosiano, S.T. Brandon and
R. Löhner

10:15 OW3 Coupled Vlasov-Maxwell Codes on Unstructured Meshes
-- A. Adoff, F. Assous, P. Degond, E. Heintzé,
F. Hermeline, P. A. Raviart, and J. Segré

10:40 break

11:00 OW4 Highly Accurate Methods for Electromagnetic Particle-in-
Cell Simulations -- A.T. Drobot

11:25 OW5 Optimizing Techniques for PIC Codes Based on
Boundary-Fitted Coordinates -- T. Westermann

11:50 OW6 A New Variational Formulation of Kinetic Plasma Theory
and the Application of Moving Finite Elements --
A.H. Glasser

12:15 OW7 A Godunov Method for the Transverse MHD Model on
Adaptive Meshes -- Z. Mikic, D.D. Schnack, I. Lottati,
S. Edelman and A. Drobot

Wednesday, September 4, 1991, 2:30pm - 4:30pm

PWA Poster Session -- Regatta Ballroom C

PWA1 Modelling Travelling Wave Output Structures with the Particle-in-
Cell Code CONDOR -- K. Eppley

PWA2 SST -- A Single-Slice Particle Simulation Code -- J. Krall, G. Joyce,
and S. Slinker

PWA3 Envelope Model of a Recirculating Heavy-Ion Accelerator --
W.M. Sharp, J.J. Barnard, and S.S. Yu

PWA4 ELBA -- A Three Dimensional Particle Simulation Code for High
Current Beams -- S. Slinker, G. Joyce, J. Krall, and R.F. Hubbard

PWA5 Nonlinear Simulation of Tokamak Turbulence: Field-Line-Following
Coordinates and Global Gyrokinetics -- A.M. Dimits

PWA6 3D Gyrofluid Simulation of Electrostatic Ion Temperature Gradient
Turbulence in a Sheared Slab -- W. Dorland and G.W. Hammett

PWA7 Binary Collision Model in Gyrokinetic Particle Simulation -- S. Ma,
R.D. Sydora, and J.M. Dawson

- PWA8 3D Electrostatic Gyrokinetic Particle Simulation of Impurity Transport -- *R.A. Santoro and W.W. Lee*
- PWA9 Plasma Formation by Beam Ionization and Collisions of Beams Focused by Such a Plasma -- *S. Rajagopalan, C.-K. Ng, and P. Chen*
- PWA10 On Solving the Self-Consistent Problem for Boundary Weakly Ionized Gas with Admixtures at Non-Linear Presentation -- *S.W. Temko, K.W. Temko, and S.K. Kuz'min*
- PWA11 Numerical Methods for Simulating Processing Plasmas -- *V. Vahedi, M. Surendra, G. DiPeso, and J. Verboncoeur*
- PWA12 Solution of Strongly Nonlinear Diffusion Equations -- *G. Bateman*
- PWA13 An Adaptive Grid with Directional Control for Toroidal Plasma Simulation -- *J.U. Brackbill*
- PWA14 3-Dimensional Eigenmode Analysis for Structures with Resistive Dielectrics and Arbitrary Boundary Conditions -- *W.A. Krueger and A.T. Drobot*
- PWA15 Extension of the IFS Toroidal Particle Code (TPC) to Include Finite Beta Effects -- *W.D. Nystrom, D.C. Barnes, M.J. LeBrun, and T.T. Tajima*
- PWA16 Computational Method for Deterministic Chaos in a Magnetically Confined Plasma -- *E.L. Vold*
- PWA17 Numerical Simulation of Resonant Alfvén Wave Absorption -- *R. Holdaway, R.A. Smith, and J.L. Geary*
- PWA18 Porting A Global Ocean Model Onto A Shared-Memory Multiprocessor: Observations and Guidelines -- *R.J. Procassini, S.R. Whitman, and W.P. Dannevik*

Wednesday, September 4, 1991, 4:30pm - 8:00pm

Free

Wednesday, September 4, 1991, 8:00pm - 10:00pm

PWE Poster Session -- Regatta Ballroom C

- PWE1 Graphical Display and Animation of Data Produced by Electromagnetic Particle-in-Cell Codes -- *M.L. Kiefer, T.D. Poinon, D.B. Seidel, R.S. Coats, and J.P. Quintenz*
- PWE2 A Post-Processor for the PEST Code -- *S. Preische, J.L. Johnson, and J. Manickam*
- PWE3 Gray Scale Rendering of Beam Profile Data -- *H. Rudd and I. Haber*
- PWE4 XGRAFIX: An X-Windows Environment for Real-Time Interactive Simulations -- *V. Vahedi and J.P. Verboncoeur*
- PWE5 Equidistribution Strategies for Particle Codes -- *W. Arter*
- PWE6 Quiet Implicit PIC -- *D.C. Barnes and R.A. Nebel*
- PWE7 Modeling the Longitudinal Wall Impedance Instability in Heavy Ion Beams Using an R-Z PIC Code -- *D.A. Callahan, A.B. Langdon, A. Friedman, D.P. Grote, and I. Haber*
- PWE8 Calculation of Gas-Discharged Plasma in Nitrogen Maintained by Microwave Electromagnetic Field -- *I.V. Bezmenov and V.P. Silakov*
- PWE9 The Stability of Non-Coplanar Rotational Discontinuities -- *P.J. Cargill and C.C. Goodrich*
- PWE10 Numerical Simulation of Electron Dynamics in Inductive Multigap Accelerators and Light Ion Beam Focusing -- *B.W. Church, D.W. Longcope, B.V. Oliver, and R.N. Sudan*
- PWE11 Comparison of 3-D VMEC and PIES Code Stellarator Equilibrium Calculations -- *J.L. Johnson, D.A. Monticello, A.H. Reiman, A. Salas, and S.P. Hirshman*

PWE12 Semi-Implicit Compressible MHD Simulation with Gravity and Temporally Second-Order Convection -- *W.S. Lawson and H.R. Strauss*

PWE13 Low n Mode Stability Analysis for $k=2$ Heliotron/Torsatron by VMEC-Step Code -- *Y. Nakamura, K. Ichiguchi, M. Wakatani, J.L. Johnson, G. Rewoldt, and S.P. Hirshman*

PWE14 Towards a Realistic Plasma Simulation Code -- *D.V. Anderson*

PWE15 Electromagnetic Modeling in Arbitrary Geometries by the Virtual Particle-Particle-Mesh Method -- *J.W. Eastwood and W. Arter*

PWE16 Solutions to Poisson's Equation on a 3D Rectangular Mesh with Internal Conductors -- *D.P. Grote, A. Friedman, A.B. Langdon, and I. Haber*

PWE17 Numerical Methods for Determination of the Charge Particle Adiabatic Motion Region in Magnetic Fields -- *S.V. Kuzmin*

PWE18 CELEST1D: An Implicit, Fully Kinetic Model for Low-Frequency, Electromagnetic Plasma Simulation -- *H.X. Vu and J.U. Brackbill*

PWE19 A High-Order Moment Simulation Model -- *K.T. Tsang, C. Kostas, D.P. Chernin, J.J. Petillo, and A. Mondelli*

PWE20 CQL(3D). A Fokker-Planck Transport Code for RF and NBI Tokamak Simulations -- *M.G. McCoy, G.D. Kerbel, and R.W. Harvey*

Thursday, September 5, 1991, 8:30am - 12:30pm

Regatta Ballrooms A & B

Special invited Talk

8:30 "The Interaction of Current Fibers in a Magnetoplasma" -- *Walter Gekelman, UCLA*

Session OT

Invited Talks -- *C. Birdsall, presiding*

9:20 OT1 A 2D Electromagnetic PIC Code for Distributed Memory Parallel Computers -- *P.C. Liewer, R.D. Ferraro, T. Krücker, V.K. Decyk, and J.M. Dawson*

9:45 OT2 3-D Electromagnetic PIC Simulation on the NRL Connection Machine -- *E. Zaldman*

10:10 OT3 Plasma PIC Simulations on MIMD Computers -- *R.D. Ferraro, P.C. Liewer, and V.K. Decyk*

10:35 break

10:50 OT4 The Numerical "Laboratory" for Plasma Simulation: A Modern Approach to the Construction and Maintenance of a Large Simulation Code -- *M.J. LeBrun, M.G. Gray, G. Furnish, T. Tajima, and W.D. Nystrom*

11:15 OT5 3D Electrostatic Gyrokinetic Simulations on Parallel Architectures -- *J.V.W. Reynders, S.S. Roy, and W.W. Lee*

11:40 OT6 Derivation of Implicit Difference Schemes by the Method of Differential Approximation -- *E.J. Caramana*

12:05 OT7 Plasma Expansion Into Vacuum -- *S. Bouquet, G. Manfredi, and M.R. Feix*

Thursday, September 5, 1991, 2:00pm - 3:30pm

Panel Discussion -- *Regatta Ballrooms A & B*

"The Transition to UNICOS and Distributed Computing: What's in it for me?" -- *A. Bland, B. Ferraro, D. Forslund, R. Zwakenberg, and B. Howard (Moderator)*

Thursday, September 5, 1991, 3:30pm - 5:30pm

PT Poster Session -- Regatta Ballroom C

- PT1 Vlasov-Maxwell Algorithm for Electromagnetic Plasma Simulation on Distributed Architectures -- *P.M. Campbell and S.A. von Laven*
- PT2 A Fully Parallel 2D Vlasov Code Comparison Between Connection Machine and Cray II Versions -- *E. Fijalkow, E. Jamin, M.R. Feix, P. Bertrand, A. Ghizzo, and M. Shoucri*
- PT3 Two Applications of the Connection Machine: The 3-D Ginzburg-Landau Equation and a 1-D Electrostatic Plasma Simulation -- *G. Furnish, L. Gil, and M.J. LeBrun*
- PT4 Numerical Solution of the Relativistic Vlasov Equation on the NRL Connection Machine -- *G. Joyce, J. Krall, and E. Esarey*
- PT5 Comparison Study: 3D Gyrokinetic PIC Codes on the Cray 2, TC2000, and CM2 -- *T.J. Williams and Y. Matsuda*
- PT6 A Multiple Species, Variable Weight Extension of the Z₅₁₀-Electron-Mass Hybrid Model -- *D.W. Hewett, G.F. Simonson, and J.A. Byers*
- PT7 A 2.5D RZ Streamlined Darwin Field Code and Its Application to Mirrortron Simulations -- *D.J. L. and D.W. Hewett*
- PT8 One Dimensional Hybrid Code Including Electron Inertial Effects -- *F. Saussède*
- PT9 An Implicit Hybrid, Quasi-Neutral Model for Low-Frequency, Electromagnetic Plasma Simulation -- *H.X. Vu and J.U. Brackbill*
- PT10 Using Adiabaticity for the Limit of Very Low Resistive Simulations -- *M.E. Kress and A.A. Blank*
- PT11 A Method for Computing Fields Near the Origin of a Cylindrical Coordinate System in Time Domain Finite-Difference Electromagnetic Simulations -- *M.F. Pasik, D.B. Seidel, M.L. Kiefer, R.S. Coats, T.D. Poinson, and J.P. Quintenz*
- PT12 Dispersion and Stability of Two Electromagnetic Solvers for Triangular Meshes -- *P.W. Rambo*

- PT13 Approximation of the Vlasov-Maxwell System Expanded in Fourier Series, by a Finite Volume Method Coupled to a Particle Simulation -- *F. Daverbon, F. Delaurens, and F. Hermeline*
- PT14 A Relativistic Eulerian Vlasov Code for the Numerical Simulation of the Laser-Plasma Beat-Wave Current Drive -- *A. Ghizzo, P. Bertrand, M. Shoucri, T. Johnston, M. Feix, E. Fijalkow, S. Karttunen, T. Patikangas, and R. Salomaa*
- PT15 Numerical Solution of the Drift-Kinetic Vlasov Equation -- *M. Shoucri, A. Ghizzo, P. Bertrand, M. Feix, and E. Fijalkow*
- PT16 Slow-Time-Scale Simulation of RF Devices -- *H.P. Freund*
- PT17 1 and 2 Species Fluid Simulations of the Solar Wind AMPTE Release -- *J.B. Harold and A.B. Hassam*
- PT18 Simulation of Luminosity Enhancement by Colliding Electron and Positron Beams in a Plasma -- *C.-K. Ng, S. Rajagopalan, and P. Chen*
- PT19 A Comparative Study of Pseudospectral and Finite Element Methods on a Velocity Shear Instability -- *S. Ghosh, R. De Fainchtein, E. Siregar, S. Zalesak, and M.L. Goldstein*
- PT20 Particle Simulations with Greatly Reduced Noise -- *M. Kotschenreuther*
- Thursday, September 5, 1991, 5:30pm - 6:30pm**
- Business Meeting -- Catamaran Room (#260)**
- Thursday, September 5, 1991, 7:00pm - 10:00pm**
- Cash Bar -- Atrium**
- Banquet -- Regatta Ballrooms A & B**
- Banquet Speaker**
- Michael F. Shlesinger, Director, Physics Division, Office of Naval Research -- "Pitfalls and Paradoxes in Probability"*

Friday, September 6, 1991, 8:30am - 1:00pm

Regatta Ballrooms A & B

Session OF

Invited Talks -- R. Faehl, presiding

8:30 OF1 MHD Boundary Layer Calculation Using the Vortex Method
-- L.A. Grizzo, K. Ganguly, A.K. Dey, and D. Shen

8:55 OF2 Eulerian Codes for Plasma Simulations -- M.R. Feix

9:20 OF3 On Enforcing Conservation Laws in Electromagnetic
Particle-in-Cell Codes -- A.B. Langdon

9:45 OF4 Two and Three Dimensional Particle-in-Cell Simulation with
MAFIA -- P. Schueff and T. Weiland

10:10 OF5 Simulation of High-Current Ion Beams in General
Accelerator Lattices -- A. Friedman, D. P. Grote, and
I. Haber

10:35 break

10:55 OF6 The ARGUS Code -- J.J. Petillo, A. Mankofsky,
A.T. Drobot, A.A. Mondelli, W.A. Krueger, and M.E. Kress

11:20 OF7 Large Space-Scale and Low Frequency Electromagnetic
Particle Simulation of Inhomogeneous Plasmas in Three-
Dimensions -- M. Tanaka, S. Murakami, H. Takamaru, and
T. Sato

11:45 OF8 An Interparticle Collision Model for Electromagnetic Hybrid
Simulations of High Density Plasmas -- M.E. Jones,
V.A. Thomas, and D. Winske

12:10 OF9 Gyrokinetic Particle Simulation of Ion Temperature
Gradient Instabilities in 3D Toroidal Geometry --
S.E. Parker and W.W. Lee

12:35 OF10 New Developments in FLIP MHD, A Particle-in-Cell Method
for Magnetohydrodynamics -- J.U. Brackbill

Wednesday Morning
September 4, 1991

Invited Oral Session (OW)
8:30am - 12:40pm

Regatta Ballrooms A & B

J. Denavit, presiding

NUMERICAL SIMULATIONS OF THE VLASOV-MAXWELL EQUATIONS BY COUPLED PARTICLE-FINITE ELEMENT METHODS ON UNSTRUCTURED MESHES.

P. Degond

CEA-Centre de Limeil-Valentou

94195 - Villeneuve St Georges Cedex France

and

Centre de Mathématiques et leur Applications / ENS-Cachan

94235 - Cachan Cedex France

1 - Introduction

In plasmas physics as well as in hyperfrequency devices or vacuum diode technology, the needs for three dimensional simulations of the Vlasov-Maxwell equations in arbitrary geometries is raising. Most of the numerical codes which are currently developed are based on finite difference approximations of the Maxwell equations on structured meshes. Such an approach is more straightforward to implement in the simple cases. However, as soon as the domain geometry becomes too complex, or when, local refinements are necessary, the structured mesh strategy requires a lot of skill, such as domain decomposition [1] or boundary fitting [2]. On the other hand, unstructured meshes provide more flexibility to approximate complex geometries, to achieve local refinements, and to eventually implement adaptive strategies. But they require a finite volume or finite element resolution of the Maxwell equations, which means, some kind of integral (or variational) formulation, together with the choice of appropriate spaces of approximation functions.

Up to now, very few finite volume or finite element methods have been developed for the numerical resolution of the Maxwell equations on unstructured meshes. The finite volume technique is described and analyzed in [4, 17]. Various finite element techniques have been tested such as conforming linear (or P1) elements [5], conforming "H(curl)" or "H(div)" elements in the standard [6], or "modified" form [7], mixed finite elements [8], or Taylor-Galerkin type methods inspired from Computational Fluid Dynamics [9, 10].

The method which we shall describe in this paper is close to [5], in that we use a decoupled second order wave equation formulation of the Maxwell equations, and a P1 conforming finite element discretization. However, in [5], no control on the divergence of the fields can be obtained. In the context of scattering or RCS calculations where no external charges and currents are present, this may not be a major drawback. The situation is opposite in plasma physics calculations where the divergence free requirements have to be included in the method. For this purpose, we propose a formulation of the Maxwell equations as a constrained problem, with associated Lagrange multipliers (which are close to the "correcting potentials" in the usual methods [11]), and we develop

a numerical approximation for both the field and the Lagrange multipliers. We use a methodology which is standard in incompressible fluid dynamics [12]. More precisely, our method is inspired from the Taylor-Hood finite element method for the Navier-Stokes equation, which has been implemented in the N3S code [13].

2 - A Taylor-Hood finite element method for the 3-D Maxwell equation [14,15,18].

We introduce the Maxwell equations as following, in the domain Ω :

$$(2.1) \quad \frac{1}{\epsilon^2} \left(\frac{\partial E}{\partial t} - \nabla \phi \right) - \nabla \times B = -\mu_0 j$$

$$(2.2) \quad \left(\frac{\partial B}{\partial t} - \nabla p \right) + \nabla \times E = 0$$

$$(2.3) \quad \nabla \cdot E = \frac{1}{\epsilon_0} \rho$$

$$(2.4) \quad \nabla \cdot B = 0$$

where $\phi = \phi(x,t)$ and $p = p(x,t)$ are the Lagrange multipliers of the constraints (2.3) and (2.4), or equivalently, the electric and magnetic correcting potentials. If ρ and j satisfy the current continuity equation.

$$(2.5) \quad \frac{\partial \rho}{\partial t} + \nabla \cdot j = 0$$

then, ϕ and p are identically zero. However, in the discrete case, it may very well happen that ϕ and p are not identically zero, either because equation (2.5) is not exactly satisfied, or because of some incompatibility of the discrete divergence and curl operators. Therefore, it is better to work with formulation (2.1) - (2.4). The problem (2.1) - (2.4) must be supplemented with appropriate boundary conditions. For the sake of simplicity, we will only consider perfectly conducting boundaries, but the case of the Silver-Müller absorbing boundary condition has been treated as well. Higher order absorbing boundary conditions are in progress. In this paper, we simply prescribe :

$$(2.6) \quad E \times n = 0$$

at the boundary $\partial\Omega$, together with homogeneous Dirichlet boundary conditions for the correcting potentials ϕ and p .

We now introduce a variational formulation of the problem (2.1) - (2.4). At the same time, we eliminate one of the fields and work with a second order vector wave equation. Let C be a vector test function in the usual space $H(\text{curl}, \Omega)$, [12] and q , a scalar test function in the Sobolev space $H_0^1(\Omega)$. Then, the formulation reads :

$$(2.7) \quad B(t) \in H(\text{curl}, \Omega), \quad p(t) \in H_0^1(\Omega), \quad \forall t \geq 0$$

$$(2.8) \quad \frac{d^2}{dt^2} (B, C) + c^2 (\nabla \times B, \nabla \times C) + (\nabla p, C) = \frac{1}{\epsilon_0} (j, \nabla \times C), \quad \forall C \in H(\text{curl}, \Omega)$$

$$(2.9) \quad (B, \nabla q) = 0, \quad \forall q \in H_0^1(\Omega)$$

Références

- [1] - A. MANKOFFSKY et al., "Domain decomposition and particle pushing for multi-processing computers", *Computer Physics Communications* **48**, (1988), pp. 155-165
- [2] - D.E. NIELSEN Jr., N. MADSEN, J. AMBROSIO, S. BRANDON, "A finite-volume particle code for Electromagnetic Plasma Simulations", *manuscript*.
- [3] - T. WEISTERMANN, "A Particle-in-Cell Method as a tool for diode simulations", *Nuclear Instruments and Methods in Physical Research* **A263** (1988), pp. 271-279
- [4] - F. HERMELINE, "Two coupled particle-finite volume methods using Delaunay-Voronoi meshes for the approximation of Vlasov-Poisson and Vlasov-Maxwell equations", *to appear in the J. Comput. Phys.*
- [5] - D.R. LYNCH and K.D. PAULSEN, "Time domain integration of the Maxwell Equations on Finite Elements", *IEEE Trans. on Antennas and Propagation*, **38** (1990), pp. 1933-1942.
- [6] - Q. LE MEUR, F. TOUZE, "Implementation of a mixed finite element in a particle method", in "Mathematical and Numerical Aspects of Wave Propagation Phenomena", G. COHEN, L. HALPERN, P. JOLY (eds), *SIAM, Philadelphia* (1991).
- [7] - J. AMBROSIO, S.T. BRANDON and R. LOHNER, "A new weighted Residual Finite Element Method for Computational Electromagnetics in the Time Domain", *preprint*
- [8] - R.L. LEE and N.K. MADSEN, "A mixed finite element formulation for Maxwell's equations in the time domain", *preprint*, Aug. 88.
- [9] - V. SHANKAR, W. HALL and A.H. MOHAMMADIAN, "A three-dimensional Maxwell's equation solver for computation of scattering from layered media", *IEEE Trans. on Magnetics*, **25**, (1989) pp. 3098-3103.
- [10] - R. LOHNER and J. AMBROSIO, "A finite element solver for the Maxwell equations", *proc. GAMNI-SMAI Conference on Numerical Methods for the Maxwell equation, PARIS, December 1989*.
- [11] - C.K. BIRDSALL, A.B. LANGDON, *Plasma Physics via Computer Simulation, Mc Graw-Hill, New York, 1985, section 15-6*.
- [12] - V. GIRAULT, P.A. RAVIART, *Finite element methods for Navier-Stokes equations. Theory and Algorithms. Springer Series in Computational Mathematics* **2** (1986).
- [13] - J.P. CHABARD, "Projet N3S de mécanique de fluides, manuel théorique, version 2.0", *rapport EDF, ref HE-4/188-09, 1988*.
- [14] - E. HEINTZE, "Résolution des équations de Maxwell par une méthode d'éléments finis 3D", *Thèse de Doctorat, in preparation*.
- [15] - F. ASSOUS, P. DEGOND, E. HEINTZE, P.A. RAVIART, J. SEGRE, "A Taylor-Hood finite element method for the 3-D Maxwell equations", *manuscript, in preparation*
- [16] - B. METIVET, B. THOMAS, P. FULLSACK, "Algorithme de gradient projeté pour la prise en compte des conditions de paroi associées aux équations de Navier-Stokes", *rapport EDF, ref Ht-726069*.
- [17] - N.K. MADSEN and R.W. ZIOLKOWSKI, "A modified finite volume technique for Maxwell's equations", *to appear in IEEE Trans. Ant. and Prop.*
- [18] - P. DEGOND, F. HERMELINE, P.A. RAVIART and J. SEGRE, "Numerical modelling of axisymmetric electron beam devices using a coupled particle-finite element method", *4th Conf ON Electr. Field Comp., Toronto (1990)*.

We obtain a similar formulation for E , except that we now need to work with the subspace $H_0(\text{curl}, \Omega)$ of $H(\text{curl}, \Omega)$ of vector fields which satisfy (2.6) at the boundary. We denote by (\cdot, \cdot) the scalar product in $L^2(\Omega)^3$; in other words :

$$(2.10) \quad (B, C) = \sum_{i=1}^3 \int_{\Omega} B_i(x) C_i(x) dx$$

Formulation (2.7) - (2.9) is a standard mixed formulation for a constrained problem. It is well known [12] that the finite element discretization of such a problem requires the choice of a pair of compatible approximation subspaces X_h and L_h respectively of $H(\text{curl}, \Omega)$ and $H_0^1(\Omega)$. The compatibility is expressed by the inf-sup (or Babuska-Brezzi) condition :

$$(2.11) \quad \sup_{C \in X_h} \frac{(C, \nabla q)}{\|C\|_{H(\text{curl}, \Omega)}} \geq \beta \|q\|_{H_0^1(\Omega)}$$

where the constant β must be independent of the mesh size h . The problem of finding such pairs of compatible approximation spaces has been well studied in incompressible fluid dynamics [12]. But among the possible choices, the Taylor-Hood element retained our attention, because, by using an appropriate quadrature formula, it leads to a diagonal mass matrix without any mass lumping. It is thus very well suited to an explicit time discretization.

We actually use the modified Taylor-Hood element (or "P₂ iso P₁") which first requires the definition of two levels of meshes. A coarser tetrahedrization is first defined, and then, a finer one is defined by dividing each tetrahedron into 8 sub-tetrahedra. Therefore, the nodes of the finer tetrahedrization consist of the vertices and of the middle of the edges of the tetrahedra of the coarser one. Then, the function space X_h contains the vector fields which are P₁ - conformal componentwise on the finer tetrahedrization. More precisely, they are continuous, and their restriction to each tetrahedron of the finer mesh is a polynomial of degree 1. At the same time, the approximation space L_h for the Lagrange multiplier p consists of the P₁ conformal finite element on the coarser grid. Therefore, much fewer degrees of freedom are needed for the Lagrange multiplier p , which is not an interesting physical quantity anyway, than for the magnetic field B . For the description of the electric field, we do not incorporate the boundary condition (2.6) in the definition of the function space, but we rather use a projection algorithm in the spirit of [16].

3 - Results

Preliminary results have been obtained in the cases of rectangular resonant cavities, rectangular waveguides, coaxial TEM modes, ... showing a very good agreement with the available analytical solutions. Other tests incorporating the treatment of net charge and currents are in progress.

Electromagnetic PIC Simulations Using Finite Elements on Unstructured Grids

John Ambrosiano and Scott T. Brandon
L-95, Box 808
Lawrence Livermore National Laboratory
Livermore, CA 94550

Rainald Löhner
CMEE School of Engineering and Applied Science
The George Washington University
Washington, DC 20052

1 Introduction

There is an ever increasing demand to use kinetic plasma simulations to model real plasma devices such as e-beam diodes, plasma torches, opening switches, and so on. With this demand comes the realization that standard simulation tools based on rectangular structured meshes are often too inflexible to accommodate real device geometries. To gain geometric flexibility, one typically must abandon structured rectangular meshes. Some have had considerable success by distorting the grid to conform to boundary-fitted curvilinear coordinates [1]. This gives substantial flexibility while retaining a logically rectangular data structure. However, to achieve the greatest flexibility, an unstructured mesh is necessary.

For the last several years, we have been engaged in designing electromagnetic particle-in-cell (PIC) simulations that can be performed on unstructured grids of triangles. We feel this a powerful strategy for matching kinetic plasma simulations to real problem geometries. Unstructured meshes not only accommodate complicated boundary shapes with ease, but also allow extreme local refinement without affecting resolution elsewhere. The basic challenges to overcome in formulating EM-PIC on these new meshes are: (a) grid generation, (b) particle interpolation and tracking, and (c) field solution. In this paper we briefly describe our techniques and will give a simple example.

2 Grid generation

For unstructured mesh techniques to be practical, grids should be automatically generated from simple descriptions of the problem geometry. We have exclusively used a technique called the advancing front method [2] which has been shown to work well. A description of the problem boundary is given in terms of straight lines or b-spline curves. This boundary curve is superimposed on a coarse background mesh of triangles. Zone size preferences are specified at the vertices of the background mesh. These values are interpolated to infer zone sizes elsewhere. The boundary is first discretized into line segments. Then, starting with the shortest segment, points are added to create triangular elements. Elements advance into the interior until the entire domain is filled. The initial mesh can then be smoothed or subrefined to produce the desired calculational mesh.

3. Particle interpolation and tracking.

To track the particles, we use a searching strategy based on the properties of linear interpolation to a triangular grid. We linearly share the charge or current density of a particle among the three vertex nodes of the triangle containing it. The function which interpolates from a given triangle vertex to another point in the domain is a linear ramp which is one at the associated vertex and zero at the other two. For a point in the interior of the triangle, all three interpolants are non-negative. However, a point outside the triangle has at least one negative interpolant. This simple property can be used to identify particles that have left their elements during the push. Because of the intrinsic shape of triangles, the element adjacent to the side opposite the node with the most negative interpolant is a good guess for where the particle has gone. The search continues until the particle is found. This process is very efficient and has been shown to vectorize well [3].

4. Field solution

The requirements of the field solution in this context are the same as those of standard finite difference techniques already in use. An appropriate method must be fast, simple to implement, robust, second-order accurate, and preserve divergence constraints to roundoff. We have developed a scheme based on finite elements that fulfills all these requirements on the meshes that we typically generate. It is a mix-d, non-conforming, non-Galerkin weighted-residual formulation described in detail elsewhere [4]. The basic idea is to use the degrees of freedom pictured in figure 1 and to pick the appropriate basis functions and weighting functions to define a good finite element method. The chosen degrees of freedom are related in an obvious way to the familiar staggered rectangular mesh schemes. The electric field in the plane is represented by its tangential projections at the midpoints of the triangle faces, whereas the magnetic field in the plane is given in terms of its normal projections. As we see in the figure, some quantities "live" on the sides, some on the vertices, and some within the element itself. The basis functions for quantities on vertex nodes and within elements are the familiar linear vertex and constant element basis functions respectively. Quantities on the sides are expressed in terms of the less familiar non-conforming linear basis functions. The fields are written in terms of their basis functions and degrees of freedom and then substituted into Maxwell's equations. These discretized equations are multiplied by appropriate weighting functions and integrated over each element to produce the finite element relations. Lumped-mass approximations are used to obtain an explicit formulation. For the TE mode the relations are:

$$B_z = -\frac{1}{\Delta_e} \sum_{j \in e} E_j \ell_j, \quad (1)$$

where B_z is the value of B_z in element e , and $j \in e$ denotes the sides j connected to e . E_j is the tangential component of E on side j . Δ_e is the element area and ℓ_j is the side length. Also,

Acknowledgement.

We wish to thank Richard DeVore, Scott Ray, Bruce Langdon, and Peter Rambo for their valuable suggestions and comments. This work was performed under the auspices of the U. S. Department of Energy by Lawrence Livermore National Laboratory under contract no. W-7405-Eng-48.

References

1. M. E. Jours, "Electromagnetic PIC Codes with Body-Fitted Coordinates," Proceedings of the 12th Conference on the Numerical Simulation of Plasmas, San Francisco, California, September, 1987.
2. R. Löhner, *Commun. Appl. Num. Methods* 4, 123 (1988).
3. R. Löhner, and J. Ambrosiano, *J. Comput. Phys.*, 91, 22 (1990).
4. J. Ambrosiano, S. T. Braundon, and R. Löhner, "A New Weighted Residual Finite Element Method for Computational Electromagnetics in the Time Domain," UCRL-JC-106408, and in *Proceedings of the 7th Annual Review of Progress in Applied Computational Electromagnetics*, Monterey, California, March 19-21, 1991.

$$\vec{E}_j = \frac{c^2}{2} \frac{(B_{j+} - B_{j-})\vec{t}_j}{(\Delta_{j+} + \Delta_{j-})/3} \quad (2)$$

where \pm refers to elements on opposite sides of side j . The $+$ element is in the direction of $\hat{z} \times \hat{r}_j$, with \hat{r}_j being the direction of the tangential E-field. For the TM mode we have:

$$\vec{B}_j = -\frac{(E_{j+} - E_{j-})}{\vec{t}_j} \quad (3)$$

Here B_j is the normal projection of \vec{B} on side j . $j \pm$ refers to vertex values on the ends of side j . The $+$ vertex is in the direction $\hat{z} \times \vec{v}_j$, with \vec{v}_j being the direction of the normal B-field. Finally,

$$\vec{E}_n = \frac{2 \sum_{j=1}^n B_j (\Delta_{j+} + \Delta_{j-}) / 3 \vec{t}_j}{\sum_{i=1}^n \Delta_i / 3} \quad (4)$$

In this case $j \pm$ refers to elements on either side of side j , and E_n is the value of E_z at the vertex node n . Leapfrog time-differencing is used to advance the above equations.

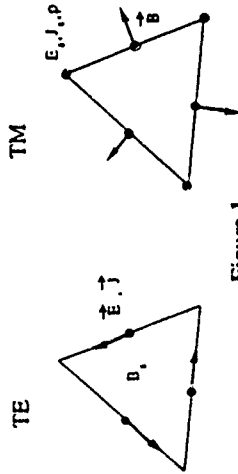


Figure 1

5 Simple example

Our computational example is an electron beam diode in x-y geometry. The grid is pictured in figure 2a. An electromagnetic (TE) wave is driven down the waveguides from the left. The pulse rises in about 0.2ns to a thereafter constant peak electric field of 10^6 V/m . The resulting potential across the approximately 0.6cm gap is about 1.5MV. Electrons are field-emitted from the boundary nodes along the curved portion of the cathode surface. After approximately 0.5ns, the self-insulated current density has the form shown in the contours of figure 2b. The result is two small arcs each carrying a current of approximately 750A across the narrowest portion of the gap. These results, requiring 2000 timesteps, were obtained in under 30 minutes on an Iris workstation model 4D-20.

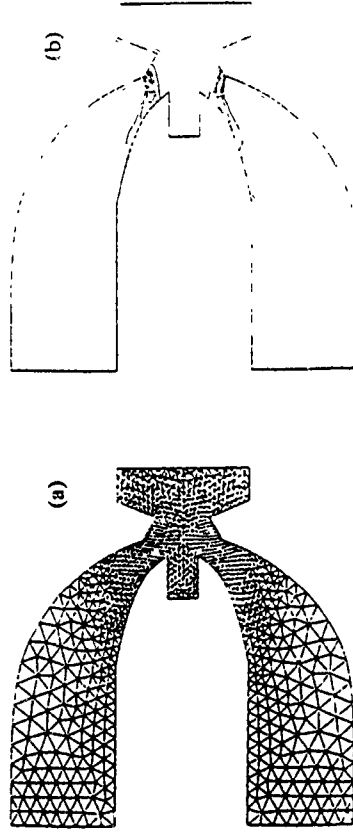


Figure 2

COUPLED VLASOV-MAXWELL CODES ON UNSTRUCTURED MESHES

by

A. Adolf, F. Assous, P. Degond, E. Heintzé, F. Hermeline, P.A. Raviart and J. Segré

Département de Mathématiques Appliquées

CEA Centre de Limeil-Valenton

94195 Villepierre St Georges - Cedex France

I - Introduction

There is a great interest in developing charged particle beam simulation methods that are well appropriate to the geometrical complexity of real devices. Several issues in the design of highly efficient FEL injectors and of high power microwave devices are crucial and lead us to prospect new techniques to approach the realistic geometries.

We have investigated two kinds of numerical methods for solving Vlasov-Maxwell equations on unstructured meshes:

- finite volume formulations on Delaunay-Voronoi meshes

- conforming finite element formulations.

We describe these methods to design prototype codes to fully 2D and 3D electromagnetic particle simulations

II - 2D finite volume approximations

To solve Maxwell equations by means of a finite volume method, we use the properties of the dual Delaunay-Voronoi meshes, where each side of a Delaunay triangle is orthogonal to a side of a Voronoi polygon [1]

To calculate B_θ and $\vec{E} = (E_r, E_z)$ in 2D-axisymmetric cases, the first method consists of:

a - integrate the Faraday law over each Delaunay triangle D with boundary ∂D

$$\frac{\partial}{\partial t} \left(\int_D B_\theta r dr dz \right) + \int_{\partial D} \vec{E} \cdot \vec{\tau} dl = 0$$

b - integrate the Ampere law over each side ∂V^k ($k = 1, N$) of the Voronoi polygons V

$$\frac{\partial}{\partial t} \left(\int_{k=1, \dots, N} \vec{E} \cdot \vec{r} dr dl \right) - c^2 \int_{\partial V^k} \vec{\nabla}(r B_\theta) \cdot \vec{\tau} dl = - \frac{1}{c_0} \int_{\partial V^k} \vec{J} \cdot \vec{r} dl$$

c - integrate the Gauss law over each Voronoi polygon of boundary ∂V

$$\int_{\partial V} \vec{E} \cdot \vec{r} dl = \frac{1}{c_0} \left(\int_V \rho r dr dz \right)$$

where \vec{r} and \vec{v} are respectively the tangential and normal unit vectors to the considered sides.

The finite volume approximation (figure 1) assumes that the tangential component $\vec{E} \cdot \vec{r}$ of the electric field $\vec{E} = (E_r, E_z)$ is constant along each side of Delaunay triangles (and the associated side of Voronoi polygons), and B_θ is constant inside each Delaunay triangle. Integrating the previous equations leads to an explicit scheme for these field components. The same calculations apply to the second set of field components E_θ and $\vec{B} = (B_r, B_z)$.

We have investigated other finite volume methods, where the Voronoi polygons and the Delaunay triangles are interchanged, as well as the degrees of freedom of the previous scheme.

III - 2D conforming finite element approximation

In order to solve Maxwell equations by means of a conforming finite element method, we introduce a variational formulation of the following problem :

$$\begin{cases} \frac{\partial^2 B_\theta}{\partial t^2} + c^2 \operatorname{curl} \operatorname{curl} B_\theta = \frac{1}{c_0} \operatorname{curl} J \\ \frac{\partial^2 E}{\partial t^2} + c^2 \operatorname{curl} \operatorname{curl} E + \operatorname{grad} \Phi = - \frac{1}{c_0} \frac{\partial J}{\partial t} \end{cases}$$

+ suitable boundary conditions (axis, perfect conductors, port conditions)

+ initial conditions

E and J denote (E_r, E_z) and (J_r, J_z) respectively, Φ is the Poisson correction potential arising from the constraint $\operatorname{div} E = - \frac{1}{c_0} \rho$ which is not exactly satisfied.

This system of equations is solved using a finite element method (figure 2) which appears to be standard in incompressible fluid flow problems: the Taylor-Hood method. Given a triangulation of the computational domain which defines the basic mesh, we introduce a refined mesh where the new vertices are the midpoints of the edges of the coarse triangles. We choose the degrees of freedom of B_θ and E as their values on the refined mesh nodes and those of Φ as their values at the basic mesh nodes.

The Mass Matrix $(\int \chi_i \chi_j r dr dz)_{i,j \leq n}$ which appears in the calculation, is diagonalized through a mass-lumping procedure which preserves the order of the scheme (χ_i is the hat function associated with each node i of the refined mesh) and leads to a very fast

explicit algorithm. E and Φ are obtained through an Uzawa's method which is classical in solving constrained problems [2]. Equivalent calculations apply to the second set of field and current components $E_\theta, B_r, B_z, J_\theta$.

Notice that the choice of this formulation results from a systematic study of various conforming finite element schemes [3].

IV - Field-particle coupling

We introduce a natural generalization to unstructured meshes of the standard assignment and interpolation procedures of the particle-in-cell method [4]. The assignment procedure, which preserves the total charge and the total current, defines the values of ρ and J at the mesh nodes :

$$\rho(\sigma_j) = \frac{q \sum_{i=1}^N w_i X_j(r_i, z_i)}{\int_0 X_j r dr dz} ; \quad \vec{J}(\sigma_j) = \frac{q \sum_{i=1}^N w_i \vec{v}_i X_j(r_i, z_i)}{\int_0 X_j r dr dz}$$

The tracking algorithm (for finding the location of a given particle in an unstructured mesh) uses interpolation results and is fully vectorized (see for example [5]).

V - 3D finite volume and finite element approximations

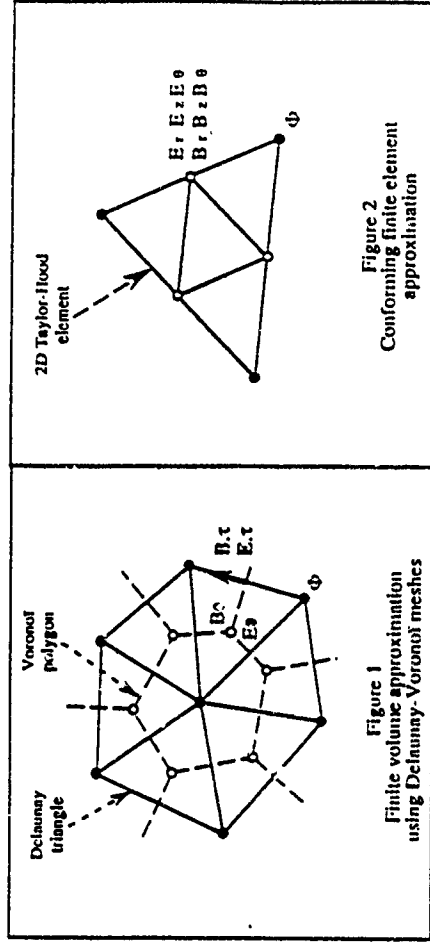
In order to deal with general three-dimensional unstructured meshes we apply the previous methods to solve Maxwell's equations in 3D cartesian geometry.

The finite volume approximation extended to the 3D case is under progress, but the difficulty to obtain 3D Delaunay-Voronoi meshes leads us to prospect and introduce corrections due to the non-orthogonality of the dual grids [6].

The development of the conforming finite element 3D Maxwell solver using Taylor-Hood elements is achieved [7]. The methodology used is quite similar to the 2D case. Good results in terms of CPU time and memory allocation are obtained and the coupling to the particles is under progress.

VI - Applications

To illustrate our work we describe the 2D simulation of electron beam devices (short bunch, photo-injector, electron guns using incoming wave ports and space charge limited particle emission) and the 3D simulation of cavity and wave guide modes.



References

- [1] - F. Hermeline, "Two coupled particle-finite volume methods using Delaunay-Voronoi meshes for the approximation of Vlasov-Poisson and Vlasov-Maxwell equations", submitted to J. of Comp. Phys.
- [2] - V. Girault and P.A. Raviart, "Finite element methods for Navier-Stokes equations", Springer Series in Computational Mathematics, N°5, Springer Verlag, Berlin, 1986
- [3] - P. Degond, F. Hermeline, P.A. Raviart and J. Segré, "Numerical modeling of axisymmetric electron beam devices using a coupled particle-finite element method", 4th Conf. on Electr. Field Comp., Toronto, 1990.
- [4] - C.K. Birdanil and A.B. Langdon, "Plasma physics via computer simulation", McGraw-Hill, New-york, 1985.
- [5] - R. Lohner and J. Ambrosiano, "A vectorized particle tracker for unstructured grids", J. Comput. Phys. 91, 22, 1990.
- [6] - N.K. Madsen and R.W. Ziolkowski, "A 3D modified finite volume technique for Maxwell's equations", LLNL Report N102802, 1990, to be published in Electromagnetics
- [7] - P. Degond, " ", this issue.

HIGHLY ACCURATE METHODS FOR ELECTROMAGNETIC PARTICLE-IN-CELL SIMULATIONS

Adam T. Drobot
Science Applications International Corporation
1710 Goodridge Drive
McLean, VA 22102

The formulation of algorithms for electromagnetic and electrostatic particle-in-cell codes has historically been biased towards simple methods that placed greater emphasis on speed and efficient use of storage than an accuracy. The codes that were developed in the late 1960's and early 1970's were forced to do so just to be tractable. The limiting factor was the speed and memory available on the CDC6600 and CDC7600 on which much of the original work was conducted.

In the late 1970's and in the 1980's the greater speed and much larger memories of CRAY like computers coupled with the success of PIC codes in attacking nonlinear kinetic problems of plasma field interaction pushed the development of codes towards applications. The investment in algorithms focused on resolving practical and fundamental issues of code stability and understanding regimes of validity. Many of the codes developed during this period continued to use variations of the basic algorithms created previously, while adding the ability to deal with a wider range of physics issues (such as particle emission, scattering materials, ionization, chemistry, etc.), improved methods for dealing with geometry, and extension to 3-D.

The continued success of electromagnetic PIC codes naturally leads to the question of how improvements can be introduced to extend their range of applicability in particular for problems where accuracy is important. In electromagnetic PIC codes this means: i) better treatment of geometry; ii) improved methods for field solvers which are correct in time and space to higher than second order and which preserve physical dispersion properties; iii) better particle orbit integration schemes; iv) conservation of physical laws; v) avoidance of numerically induced noise and instabilities; and vi) improved treatment at boundaries.

In this paper I describe a systematic approach for constructing highly accurate methods using both finite difference and finite element techniques. The methods are based on unstructured grid solvers for the fields. The particle orbit integrators include a local field expansion which is tightly coupled to the field solver. The order and accuracy of the schemes will be illustrated from simple examples.

Optimizing Techniques for PIC Codes based on Boundary-Fitted Coordinates

Thomas Westermann

Kernforschungszentrum Karlsruhe GmbH, Abteilung für Numerische Physik, HDI-NP
P.O. Box 3640, 7500 Karlsruhe, F.R.G.

Abstract. For simulating complicated technical geometries sophisticated techniques were developed to provide convenient flexibility in fitting the mesh points on the boundary. Experience shows that it is not sufficient to treat only the geometrical influences accurately. Improvements in the numerical solution are also required. They can be obtained either by using higher-order approximations or by changing the grid with respect to the numerical solution. Both methods will be discussed in the special case of the BFCPIC code.

1. Introduction

Particle-in-cell codes are an attractive computational tool for studying kinetic phenomena e.g. in plasma physics [1]. For simulating complicated technical geometries, where the shape of the boundary quantitatively determines the physical processes occurring inside these electromagnetic devices, several techniques were developed to take the shape of the boundary into account accurately:

One approach is to use unstructured meshes and finite element techniques [2], and another one is to use structured non-regular 4-point meshes fitted to the boundary and finite difference methods [3,4]. Both methods provide convenient flexibility in fitting the mesh points on boundaries. However, it turns out, that it is not sufficient to treat only the geometrical influences accurately and choose the grid points in the interior as a good initial guess for the beginning of the simulation. Improvements in the numerical simulation are required. They can be obtained with different approaches: Either by changing the solution method to a higher-order approximation or by changing the grid with respect to the numerical solution. Since the solution of the problem is not known a priori, the grid must dynamically be adapted to the numerical solution during the simulation. This leads to the concept of adaptive mesh generation [5,6].

Both methods are investigated for the special case of the BFCPIC code [4] (Particle-in-Cell code based on Boundary-Fitted Coordinates). The BFCPIC code is an electro- and magneto-static 2 1/2-D PIC code in cylindrical coordinates (z, r) using extended finite difference methods on structured curvilinear meshes. In the past, numerical simulations were being performed successfully with the BFCPIC code in order to optimize pulsed power ion diodes [7].

2. Grid generation

We follow the grid generation techniques introduced by Thompson [8] via elliptic partial differential equations: Let $\xi(x, y)$, $\eta(x, y)$ be the mapping of the physical area in the (x, y) -plane onto the logical area in the (ξ, η) -plane given as the solution of the elliptic system

$$\Delta \xi = P, \quad \Delta \eta = Q.$$

Then the inverse map $x(\xi, \eta)$, $y(\xi, \eta)$ satisfies

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} + \gamma^2 x_{\xi} P + \gamma^2 x_{\eta} Q = 0$$

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} + \gamma^2 y_{\xi} P + \gamma^2 y_{\eta} Q = 0$$

where

$$J := x_{\xi} y_{\eta} - y_{\xi} x_{\eta}$$

and

$$\alpha := x_{\eta}^2 + y_{\eta}^2, \quad \beta := x_{\xi} x_{\eta} + y_{\xi} y_{\eta}, \quad \gamma := x_{\xi}^2 + y_{\xi}^2.$$

Solving the discretized Dirichlet problem with $P = 0$ and $Q = 0$ leads to the grid points (x_{ij}, y_{ij}) in the physical space.

3. Improvements of the numerical solution

Experience with our stationary assumption ($E = -\text{grad } \Phi = -(\Phi_{\xi}, \Phi_{\eta})$, $\Delta \Phi = -\rho/\epsilon$) showed that numerical errors accumulate in the numerical differentiation:

$$\Phi_i = (r_{\eta} \Phi_{\xi} - r_{\xi} \Phi_{\eta}) / J, \quad \Phi_j = (-x_{\eta} \Phi_{\xi} + x_{\xi} \Phi_{\eta}) / J.$$

We could considerably improve the numerical differentiation in the logical coordinate system: The standard 2-point difference scheme for point (i, j)

$$\Phi_{\xi} = 0.5(\Phi_{i+1,j} - \Phi_{i-1,j}) / \Delta \xi, \quad \Phi_{\eta} = 0.5(\Phi_{i,j+1} - \Phi_{i,j-1}) / \Delta \eta,$$

was replaced by a non-equidistant 3-point scheme by taking into account the local scaling of the intersection of the grid points $\mathbf{x} := (x_{i-1,j}, y_{i-1,j})$, $(x_{i,j}, y_{i,j})$, and $\mathbf{x}^* := (x_{i+1,j}, y_{i+1,j})$:

$$\Phi_{\xi} = (k^+ \Phi_{i-1,j} + k^0 \Phi_{i,j}^* + k^- \Phi_{i+1,j})$$

with

$$h^- = -\frac{h_2}{h_1(h_1 + h_2)}, \quad h^0 = \frac{h_2 - h_1}{h_1 h_2}, \quad h^+ = \frac{h_1}{h_2(h_1 + h_2)},$$

and

$$h_1 = \|x^0 - x^+\| \quad \text{and} \quad h_2 = \|x^0 - x^-\|$$

Here, x^0 reads as the projection of the vector (x_{ij}, y_{ij}) on the vector $x^+ - x^-$ and Φ^0 is an interpolated value of Φ onto point x^0 . An analogous formula is valid for Φ_η .

Further improvements can be achieved by adapting the grid to the numerical solution. For this purpose mainly two techniques are introduced: The first is to redistribute a fixed number of points from regions of relatively small error or small gradients to regions of large errors or large gradients in choosing the functions P and Q in a proper way. The second method is to add grid points in regions of large errors or gradients.

An estimate for the relative error of the difference scheme is given in the logical space by

$$F_{rel} = 1/8 \cdot h^3 / \epsilon \cdot \rho_{tt}$$

when h is an average value of the local scaling in the grid spacing.

This expression is used in the two-dimensional approach for ξ and η to define the functions P and Q , respectively. Instead of an estimate for the error in the difference scheme, the gradient of the charge density of the ions and electrons can be used for the redistribution of the grid points.

A second method is to add grid points in regions of large errors or gradients. As in the latter case criteria for inserting grid points are given by error estimates or by the gradient of the densities.

4. Results

Both techniques were implemented in the BFCPIC code. As an example the case of a self magnetically insulated ion diode will be discussed. Fig. 1 shows a boundary-fitted grid for the self-magnetically insulated Bg-diode [9]. The grid points in the interior are distributed with respect to the empty diode configuration without charged particles. Figs. 2 and 3 show grids fitted to the gradient of the charge density of the ions and electrons, respectively. It will be shown that such grid adaption leads not only to a better resolution of the physical problem but also to a smoothing of the noise properties involved in PIC computations.

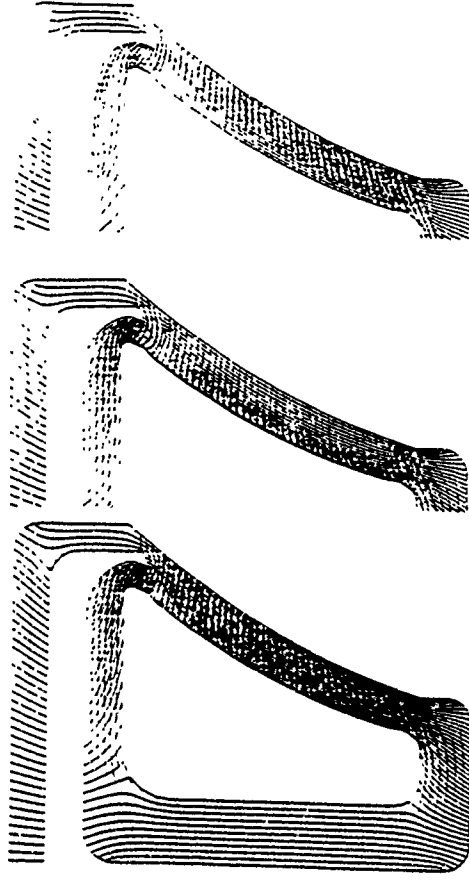


Fig 1.

Fig 2

Fig 3

Acknowledgements

The author wishes to acknowledge the stimulating discussions with M. Alef, S. Illy and Prof. W. Schmidt and the carefully testing of the algorithm by A. Müller

References

- [1] R.W. Hockney, J.W. Eastwood, Computer Simulation Using Particles, McGraw-Hill, 1981.
- [2] R. Löhner, J. Ambrosiano, A Vectorized Particle Tracer for Unstructured Grids, J. Comp. Phys. 91, 22-31, 1990.
- [3] M.E. Jones, Electromagnetic PIC Codes with Body-Fitted Coordinates, Proc. 12th Conf. Num. Sim. of Plasmas, San Francisco, Sept. 20-24, 1987.
- [4] T. Westermann, A Particle-in-Cell Method as a Tool for Diode Simulations, Nucl. Instr. Methods A263, pp. 271-279, 1988.
- [5] H.J. Kim, J.F. Thompson, Three-Dimensional Adaptive Grid Generation on a Composite-Block Grid, AIAA Journal 28 (3), 470-477, 1989.
- [6] J.U. Brackbill, J.S. Saltzman, Adaptive Zoning for Singular Problems in Two Dimensions, J. Comp. Phys. 46, 342-368, 1982.
- [7] T. Westermann, Optimization by Numerical Simulation of the Focusing Properties of Self-Magnetically Insulated Ion Diodes, Appl. Phys. Lett. 58 (7), 696-698, 1991.
- [8] J.F. Thompson, Z.U.A. Warsi, C.W. Mastin, Boundary-Fitted Coordinates, Systems for Numerical Solution of Partial Differential Equations - A Review, J. Comp. Phys. 47, 1-108, 1982.
- [9] W. Schimasek, W. Bauer, O. Stoltz, Generation of a Focused Proton Beam with a Selfmagnetically Bg-Insulated Ion Diode, Rev. Scientific Instr. 62 (1), 168-177, 1990.

A New Variational Formulation of Kinetic Plasma Theory and the Application of Moving Finite Elements

Alan H. Glasser, Los Alamos National Laboratory
14th International Conference on Numerical Simulation of Plasmas
Annapolis, MD, September 3-6, 1991

A new variational formulation has been developed for the system of equations governing kinetic plasmas and electromagnetic fields. It is used to apply the method of Moving Finite Elements (MFE)¹⁻³ to the electromagnetic fields. The fields are expanded in a basis of linear finite elements on a movable, unstructured grid of triangles in 2D or tetrahedra in 3D, while the plasma distribution function is expanded in a basis of super particles. Minimization of the variational with respect to the time derivatives of the field quantities yields a coupled system of equations for simultaneously advancing the amplitudes and node positions, resulting in adaptive grid motion. The adaptivity of the grid may save a large factor in the size of the grid and the number of particles required in many problems. Minimization of the variational with respect to the time derivatives of the particle positions and velocities gives the equations of motion, providing consistent prescriptions for assigning particles to the grid and fields to the particles. Orthogonality conditions on the particles are derived as conditions for keeping their equations of motion independent. Collisions can be included in a natural way. The relationship between PIC methods and alternative methods of discretizing phase space is clarified.

The method starts with three fundamental equations. Two of these are the Maxwell equations containing time derivatives, i.e. Ampère's Law and Faraday's Law,

$$\partial \mathbf{E} / \partial t = c \nabla \times \mathbf{B} - 4\pi \mathbf{j}, \quad \partial \mathbf{B} / \partial t = -c \nabla \times \mathbf{E}. \quad (1)$$

The remaining field equations are either initial conditions or derivable from the fundamental ones. The other fundamental equations are the kinetic equations for the particles,

$$\frac{\partial f_j}{\partial t} + \mathbf{v} \cdot \frac{\partial f_j}{\partial \mathbf{x}} + \frac{c_L}{m_j} \left(\mathbf{E} + \frac{1}{c} \mathbf{v} \times \mathbf{B} \right) \cdot \frac{\partial f_j}{\partial \mathbf{v}} = \dot{C}_j \quad (2)$$

There are several species, denoted by the index j . Note that this equation allows for an arbitrary collision term \dot{C}_j , e.g. Boltzmann or Fokker-Planck, which may be dropped if desired to obtain the collisionless Vlasov equation.

The variational is obtained by taking each of the principle equations, squaring it, multiplying by a weight function to be determined later, and integrating. For the field equations, the integral is over the spatial domain, while for the kinetic equations the integral is over phase space and there is a sum over species. Thus,

1

$$L = L_E + L_B + L_J,$$

$$L_E = \int d^3x [\partial \mathbf{E} / \partial t - c \nabla \times \mathbf{B} + 4\pi \mathbf{j}]^2 w_E,$$

$$L_B = \int d^3x [\partial \mathbf{B} / \partial t + c \nabla \times \mathbf{E}]^2 w_B,$$

$$L_J = \sum_j \int d^3x \left[\frac{\partial f_j}{\partial t} + \mathbf{v} \cdot \frac{\partial f_j}{\partial \mathbf{x}} + \frac{c_L}{m_j} \left(\mathbf{E} + \frac{1}{c} \mathbf{v} \times \mathbf{B} \right) \cdot \frac{\partial f_j}{\partial \mathbf{v}} - C_j \right]^2 w_J. \quad (3)$$

Equations (1) and (2) are recovered from the variational principle by minimizing with respect to the time derivatives of the variables,

$$\frac{\delta L}{\delta (\partial \mathbf{E} / \partial t)} = \frac{\delta L}{\delta (\partial \mathbf{B} / \partial t)} = \frac{\delta L}{\delta (\partial f_j / \partial t)} = 0. \quad (4)$$

This variational is different from most such quantities used in plasma physics. It is neither an energy, as in ideal MHD, nor an action, as in Eastwood's finite element PIC formulation,⁴ but the residual or error in solving the equations by approximation methods, as in the application of Moving Finite Elements to fluid equations.

For the electro-magnetic fields, we use linear finite elements on a movable, unstructured grid of triangles in 2D and tetrahedra in 3D. Simultaneous minimization with respect to the time derivatives of the amplitudes and node positions generates a coupled system of equations for advancing these quantities in time, equivalent to the MFE method applied previously to fluid equations. In response to these equations, the grid goes where it is needed to resolve the evolving solution. The combination of an implicit time step and the smooth motion of the grid and fields together is effective at overcoming the CFL limit on the time step size, allowing it to advance a moving front by many front thicknesses in one step. The grid can be occasionally reconnected to allow it more flexibility in fitting the topology of the solution. The field equations could be supplemented by moment equations to improve the accuracy of an implicit time step advance, as in implicit moment methods,⁵ with the resulting information about moments then discarded in favor of the more precise results of the particle advance. The fluid quantities would then also be expanded in Moving Finite Elements and would contribute to the determination of the grid motion.

When applied to problems with narrow boundary layers, such as sheaths and tearing modes, the adaptivity of the grid could save a large factor in the size of the grid required. For example, if the width of a low- n tearing mode in a tokamak is determined by an ion gyration radius while the separation mode rational surfaces is on the order of the minor radius of the device, this factor may be of order 10^3 . Since the number of particles required is primarily determined by acceptable noise levels, a reduction in the grid size could save a comparable factor in the number of particles required. These savings could make a make contribution to the feasibility of the Numerical Tokamak project.

2

Super particles are introduced as basis functions in phase space for expansion of the distribution function to obtain a Galerkin method,

$$f_j(\mathbf{x}, \mathbf{v}, t) = \sum_i n_{i,j} f_i(\mathbf{x} - \mathbf{x}_{i,j}(t), \mathbf{v} - \mathbf{v}_{i,j}(t)). \quad (5)$$

The basis functions f_i may be thought of as rigid clouds in phase space, with density normalized to unity and then multiplied by the number of particles $n_{i,j}$ in each super particle. For the collisionless case, minimization of the variational with respect to the time derivatives of the super particle positions and velocities gives the equations of motion,

$$\begin{aligned} \frac{1}{2} \frac{\delta L_j}{\delta \dot{\mathbf{x}}_{i,j}} \frac{d}{dt} &= n_{i,j}^2 \int d^3 \mathbf{x} \int d^3 \mathbf{v} w_j \frac{\partial f_j}{\partial \mathbf{x}} \cdot \left(\frac{d\mathbf{x}_{i,j}}{dt} - \mathbf{v}_{i,j} \right) = 0, \\ \frac{1}{2} \frac{\delta L_j}{\delta \dot{\mathbf{v}}_{i,j}} \frac{d}{dt} &= n_{i,j}^2 \int d^3 \mathbf{x} \int d^3 \mathbf{v} w_j \frac{\partial f_j}{\partial \mathbf{v}} \cdot \left\{ \frac{d\mathbf{v}_{i,j}}{dt} - \frac{e_j}{m_j} \left[\mathbf{E}(\mathbf{x}_{i,j} + \mathbf{x}) + \frac{1}{c} \mathbf{v} \times \mathbf{B}(\mathbf{x}_{i,j} + \mathbf{x}) \right] \right\} = 0. \end{aligned} \quad (6)$$

The first of these is the equation for the velocity, multiplied by a positive-definite matrix. The second is the equation for the acceleration, appropriately averaged over the spatial extent of the super particle. The combination of these equations of motion with the discretized field equations discussed above gives a consistent prescription for weighting the fields on the particles and the particles on the fields.

When the expression for f in Eq. (5) is introduced into the variational in Eq. (3), the sum over super particles occurs inside the square, leading to cross terms which couple the equations for different super particles. In order to eliminate these cross terms and obtain the independent equations for each super particle in Eq. (6), conditions must be imposed on the super particles. While it is sufficient to make $f_i(\mathbf{x}, \mathbf{v})$ an even function of \mathbf{x} and \mathbf{v} , and to make the super particles be nonoverlapping in phase space, these conditions are excessively restrictive. Necessary and sufficient conditions can be expressed as orthogonality properties in phase space. The physical interpretation of these orthogonality conditions can be understood as follows. It is well known that the gradients of the distribution function in velocity space give rise to phase mixing and to Landau damping and growth. The choice of rigid clouds in phase space legislates that the phase mixing of a single particle is excluded from the treatment. However, different overlapping particles can influence each other's motion through a kind of remnant of phase mixing, unless this is explicitly excluded by the orthogonality conditions. While this interaction could in principle be retained, it would substantially increase the difficulty of advancing the particle equations.

There remain some unsolved problems associated with these orthogonality conditions. If a given distribution of super particles satisfies the orthogonality conditions initially, it is not clear that they satisfy them later in time, after following the trajectories specified

by the equations of motion obtained from the variational principle. In the limit of delta functions, i.e. point particles, it is well known from Hamiltonian dynamics that particles which are initially distinct in phase space never collide, proving the point. Whether this property is preserved for finite-sized particles has not been proven. In addition, the inclusion of a collision term would almost certainly compromise this. This suggests that PIC methods may have a problem describing systems with both collisions and Landau damping or growth.

Once the variational principle is used to understand particles as basis functions in a Galerkin expansion, other sets of basis functions may be considered and compared to them. For example, it would be possible to discretize phase space with 5D (for gyrokinetic problems) or 6D (for full dynamics) simplices on a moving grid and thus apply Moving Finite Elements to phase space as well as configuration space. This might, for example, be useful in describing trapped particle modes, which have sharp gradients in the distribution function at the boundary in velocity space between trapped and untrapped particles. It would also avoid the need to keep track of the relative positions of particles and grid cells. However, its usefulness would be limited by the fact that the Jacobian matrix would be dense rather than sparse in velocity space due to the global coupling of finite elements by the integrals required to obtain current densities. The well-known nearest-neighbor interaction of finite elements applies only to the discretization of partial differential equations, whereas Maxwell-Vlasov equations form a system of integro-differential equations. Another choice of basis is a sum of a Maxwellian plus super particles, yielding a form of δf method. Other systems of basis functions with additional desirable properties might be devised to discretize phase space, using the variational principle introduced here.

References

1. K. Miller and R. N. Miller, "Moving Finite Elements I," *SIAM J. Num. Anal.* **13**, 6, 1019 (1981); K. Miller, "Moving Finite Elements II," *SIAM J. Num. Anal.* **18**, 5, 1033 (1981).
2. Neil Carlson and Keith Miller, "Gradient Weighted Moving Finite Elements in Two Dimensions," in *Proceedings of the ICASE/NASA Workshop on Theory and Application of Finite Elements*, July 28-30, 1986, Hampton, Virginia, published by Springer-Verlag.
3. Alan H. Glasser, "A Moving Finite Element Model of the High-Density Z-Pinch," *J. Comp. Phys.* **85**, 159 (1989).
4. James W. Eastwood, "The virtual particle electromagnetic particle-mesh method," *Comp. Phys. Comm.* **64**, 252 (1991).
5. J. U. Brackbill and D. W. Forslund, "Simulation of Low-Frequency Electromagnetic Phenomena in Plasmas," in *Multiple Time Scales*, ed. J. U. Brackbill and B. I. Cohen (Academic Press, New York, 1985), p. 272.

A GODUNOV METHOD FOR THE TRANSVERSE MHD MODEL ON ADAPTIVE MESHES

Z. Mikic, D. D. Schnack, I. Lottati, S. Eldelman, and A. Drobot
Science Applications International Corporation
San Diego, CA and McLean, VA

ABSTRACT

The Godunov method, and its higher-order extensions, has been very successful in treating flows with shock waves in gas dynamics. These algorithms for the inviscid Euler equations are based on exact local solutions to the Riemann problem. They have the desirable properties of being able to describe shock waves and contact discontinuities while preserving monotonicity in the physical variables (i.e., they are non-oscillatory). We have extended the second-order Godunov method to the transverse MHD equations, in order to simulate plasmas with small viscosity and resistivity. In the case of the transverse MHD model (a 2D system in which the magnetic field is transverse to the flow), the Riemann problem can be solved exactly. The second-order Godunov method, combined with adaptive, unstructured grids, produces a very robust algorithm for transverse MHD problems. Shocks, discontinuities, and current sheets can be modeled in complicated geometries without special shock-tracking. The development of a 2D code which is based on triangular mesh elements, as well as its application to impulsive plasma phenomena, will be described.

UNSTRUCTURED ADAPTIVE MESHES

Problems which have shocks require the resolution of multiple length scales. Unstructured meshes have proven to be useful for the simulation of such problems in physical domains with complicated geometry [1]. Unstructured meshes allow the computational mesh to be selectively refined and coarsened, thereby permitting the mesh to adapt dynamically as the simulation proceeds. In 2D problems, we have used triangles as basis

elements in our unstructured grids. Refinements of 100:1 in length scales are used routinely.

The mesh is adapted automatically using a scheme based on density gradients and the flow of mass and energy within simulation cells. The scheme does not require explicit "shock tracking." Cells in which refinement and coarsening is indicated are flagged according to specified criteria. Mesh points are added in "buffer zones" around cells flagged for refinement in order to "anticipate" shocks.

THE TRANSVERSE MHD MODEL

In this model, the flow is transverse to the magnetic field. In Cartesian coordinates, if we assume that the flow is in the x - y plane, the flow velocity is defined as $\mathbf{v} = u(x,y)\hat{x} + v(x,y)\hat{y}$ and the magnetic field is given by $\mathbf{B} = B(x,y)\hat{z}$. In this case, the ideal MHD equations can be written as:

$$\frac{\partial U}{\partial t} + \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} = 0, \quad (1)$$

where

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ B \end{bmatrix}, \quad F_x = \begin{bmatrix} \rho u \\ \rho u^2 + p^* \\ \rho u v \\ B u \end{bmatrix}, \quad F_y = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p^* \\ B v \end{bmatrix}, \quad (2)$$

where \mathbf{U} is the vector of conserved fluid quantities, F_x and F_y are the x and y fluxes, p^* is the total pressure, $p^* = p + B^2/2$, and e is the specific energy, $e = p/(\gamma-1) + B^2/2 + \rho(u^2+v^2)/2$. In axisymmetric cylindrical coordinates the transverse MHD model corresponds to a situation in which the flow is poloidal and the magnetic field is toroidal (i.e., z -pinch geometry).

THE GODUNOV METHOD

The Godunov method [2] is a member of the class of conservative upwind difference schemes. It is popular in hydrodynamics, and is the basis of the PPM method of Colella and Woodward [3]. Consider a representative mesh triangle with index i , and its neighbors, as shown in Fig. 1. The

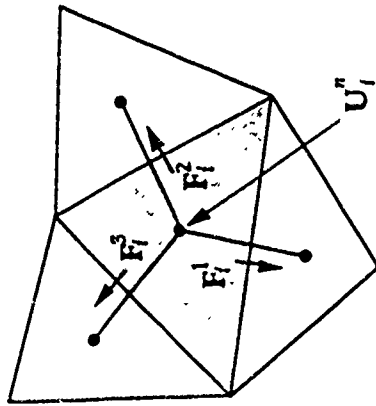


Fig. 1. Representative triangular cell in the mesh, showing the fluxes across the edges.

solution U_i^n in cell i at time step n is defined at the center of the triangle. In order to advance to the next time step, Eq. (1) is differenced as:

$$(U_i^{n+1} - U_i^n) / \Delta t + \sum_{j=1}^3 F_j^i / l_j^i / A_i = 0$$

where F_j^i is the flux out of cell i across edge j , l_j^i is the length of edge j , and A_i is the area of cell i . In the first-order Godunov method [2], the fluxes F_j^i are determined by solving local 1D Riemann problems with left and right states given by the states U_i^n at the centers of cells lying to the left and right of edge j . These solutions to local Riemann problems are used to compute the fluxes F_j^i according to Eq. (2). The original Godunov method is first-order accurate in Δt and Δx . A second-order extension [1,4] is obtained by using gradients to estimate second-order-accurate guesses to the left and right states for Riemann problems across cell edges. In these extrapolations it is necessary to apply monotonicity and "limiter" constraints based on the characteristic equations [3]. The second-order method yields much superior resolution of shocks at a modest increase in complexity.

APPLICATION

The method has been applied to a plasma gun problem. In which a uniform plasma is subjected to an impulsive electric field. This generates a contact discontinuity and a shock wave which travels into the plasma, interacts with the surrounding walls, and is accelerated out of the simulation domain. Figure 2 shows selected snapshots of the grid during the evolution of the plasma, illustrating how the grid adapts dynamically as the solution evolves.

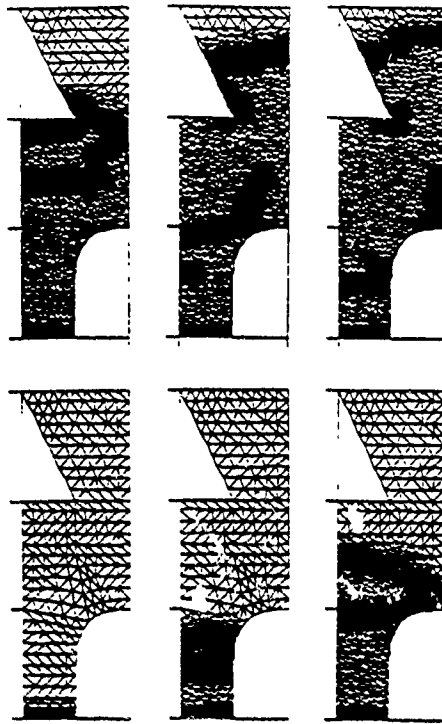


Fig. 2. Snapshots of the dynamically adapting mesh.

REFERENCES

- [1] I. Lottati, S. Eidelman, and A. Drobot, "A Fast Unstructured Grid Second Order Godunov Solver," 28th Aerospace Science Meeting, AIAA-90-0699, (1990).
- [2] S. K. Godunov, *Mat. Sborn.* 47, 271 (1959).
- [3] P. Colella and P. R. Woodward, *J. Comput. Phys.* 54, 174 (1984).
- [4] S. Eidelman, P. Colella, and R. P. Shreeve, *AIAA Journal*, 22, 1609 (1984).

Wednesday Afternoon

September 4, 1991

Poster Session (PWA)
2:30pm - 4:30pm

Regatta Ballroom C

MODELLING TRAVELLING WAVE OUTPUT STRUCTURES WITH THE PARTICLE-IN-CELL CODE CONDOR*

Kenneth Eppley
Stanford Linear Accelerator Center
PO Box 4349, Stanford, California, 94309
ABSTRACT

Multimegawatt klystrons at X band or higher frequencies can have field gradients in the output stage high enough to cause breakdown if applied across a single gap. One solution is to replace the output cavity with a travelling wave structure in which the interaction is extended over a longer distance. An added benefit is higher efficiency than is possible with a single gap. However, the strong beam loading makes linear travelling wave theory a poor approximation. Thus we are developing algorithms for non-linear, self-consistent treatment of travelling wave structures interacting with electron beams.

We are using the Particle-in-Cell code CONDOR [1]. We have developed two algorithms. The first models the actual cavity geometries and is useful for simple, two-dimensional structures such as disk-loaded waveguides with low external Q and relatively high group velocity. In our previous klystron simulations [2], we did not attempt to model the actual cavity geometry because, first, the high Q of many cavities causes very long fill times, and, second, very fine zoning would be required to model the cavity resonant frequency to adequate accuracy. Many disk loaded output structures of interest, however, have Q of order 10 to 20, and group velocities of 1 c or more. Transient times are then tens of RF periods rather than hundreds. The low Q also makes the induced voltage less sensitive to the exact cavity frequency. This plus the simplicity of the structure permits coarser zoning to be used. As a result, the computational time is low enough to use this algorithm for design studies. For some test problems of practical interest we have found execution times to be a factor of two to four (depending on zoning) what would be required for standing wave cavities using the port algorithm. If one uses the coarsest acceptable mesh, the dispersion curve can be shifted in frequency by about a percent from the true values, which must be kept in mind when relating the results to actual designs.

The klystron preprocessor code has been adapted to make it easy for users to run problems with disk-loaded output sections. The user need only specify the disk separations and inner and outer radii. Rounded ends to the disks are available as an option (However they require about twice the number of mesh points to get an accurate frequency as do cells with flat ends.)

Cells may be loaded by waveguide coupling. This is modelled by using a port at the outer radius with a specified transmission coefficient. The Q of the cavity varies inversely with this value. A simulation must be done initially to determine the proportionality constant. Fortunately, it is rather insensitive to cavity dimensions.

* Work supported by the US Department of Energy, contract DE-AC03-76SF00515

Full modelling of the cavities has several advantages compared to the port approximation. First, coupling between cells is treated properly. Second, fields across the gaps are treated more accurately. Third, higher order modes are included. (However, the present model will not have the right Q for a higher mode if a coupler is specified. This could be solved, if necessary.) The limitations of this model compared to the port approximation are that first, it is more expensive, especially for more complicated structures or structures with low group velocity, and second, it cannot model non-axisymmetric slot-coupled cavities. We have developed an alternate formalism, using the port model, which is an extension of the impedance matrix model we used for standing wave structures. We verified that the port approximation gives good agreement with the full cavity simulation, at least when the beam does not get too close to the walls. For a test case we took the gap voltages and phases found for the full cavity model and imposed them as port voltages with the same input beam. The resulting induced currents agreed well with the cavity model.

The impedance matrix model is based on the single gap port model described elsewhere [2]. Briefly, we seek a steady state solution for the complex voltage $V(t)$ which satisfies

$$V(t) = Z I(t) \quad (1)$$

where I is the complex induced current and Z is the impedance. We form a time dependent equation for the voltage:

$$dV(t)/dt = -k [V(t) - Z I(t)] \quad (2)$$

$V(t)$ will adjust itself until the impedance relation is satisfied self-consistently. The induced current is computed from the integral of $E \cdot J$. The equation converges faster if one takes into account the beam loading, assuming that the change in induced current is a linear function of the change in voltage, i.e.

$$\Delta I = \alpha I \cdot \Delta V \quad (3)$$

Then

$$\Delta V = -k \Delta t (V - Z I) / (1 - \alpha Z) \quad (4)$$

To model a general impedance matrix we replace the term $Z I$ in Equation (4) by $\sum Z_{ij} I_j$. The problem is run until a steady state solution is obtained. When the beam starts off highly modulated, as in an output section, the method converges readily

For a structure which has not been built, we need a method to calculate the coefficients of the matrix. For an axisymmetric structure one can use CONDOR without beam, driving the structure with a voltage or an antenna. As a test case we used one of the disk-loaded structures modelled with the full cavity method. To calculate the coefficients we drove each gap in turn with a small antenna centered in the cavity gap. (The currents for the non-driven gaps were so much smaller that they could be ignored. Thus we could obtain the matrix elements by inspection from the induced voltages. Computing the matrix elements thus required one time-dependent simulation for each cell, run to steady state. The total time used was comparable to the full cavity simulation including beam. Of course, once one has the matrix, one can run the same structure many times for substantially less cost. In principle, a frequency domain code could calculate the matrix elements more cheaply. Code development in this area is needed.

The application of the impedance matrix to the same problem solved with the full cavity model gave good agreement. The comparison between voltages and phases for a four cell structure calculated with both methods is shown in Figure 2

For three dimensional slot-coupled structures, the matrix formalism should still be applicable. The field in the drift tube is still axisymmetric to a good approximation, so we can use the port approximation with CONDOR. This method has been applied to slot-coupled standing wave structures with reasonable success [3]. One would need to use a code such as ARGUS, MAFFIA, or SOS to calculate the matrix elements. One would run the code in the time domain and drive the system with an antenna. One needs to obtain the RF voltage and phase and induced current and phase across each cavity gap. (The induced current is equal to the average of the magnetic field across the gap.) If one can adjust the antenna such that the induced current in each gap is much larger than those in the other gaps, one can read off the matrix elements from the induced voltages. Otherwise, one must repeat the calculation a number of times, shorting out each of the gaps in turn, until one has enough sets of data to solve for the coefficients of the matrix.

References

- 1 CONDOR User's Manual, B. Aimineti, S. Brandon, K. Dyer, J. Moura, D. Nielsen, Jr., Livermore Computing Systems Document, Lawrence Livermore National Laboratory, Livermore, California, April, 1988.
- 2 K. Eppley, "Algorithms for the Self-Consistent Simulation of High Power Klystrons," SLAC PUB 4622, Stanford Linear Accelerator Center, May 1988.
- 3 K. Eppley, W. Herrmannsfeldt, T. Lee, "Improving the Efficiency of Microwave Devices with a Double Output Cavity," SLAC-PUB-3984, May, 1986.

Figure 1. Electron position space distribution through a four cell disk-loaded travelling wave structure. Horizontal axis is Z and vertical axis is R (not to scale). The input beam had been bunched by a gain section in a previous simulation (not shown). The first and last cell were loaded down by coupling to output ports to give $Q = 10$

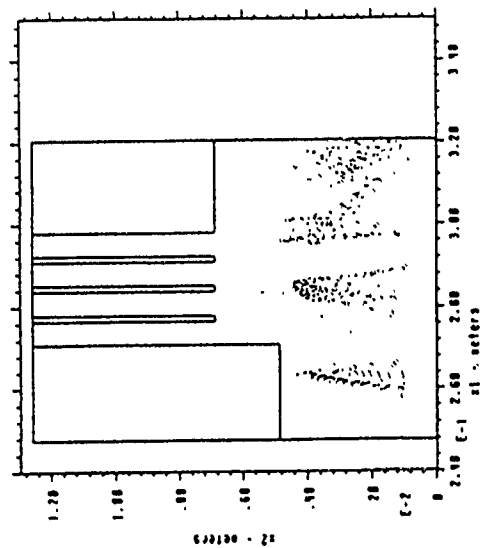
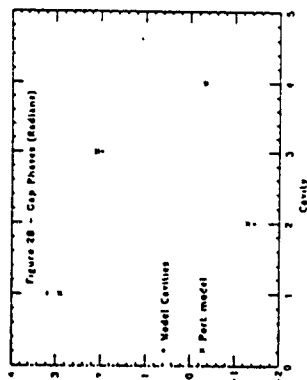
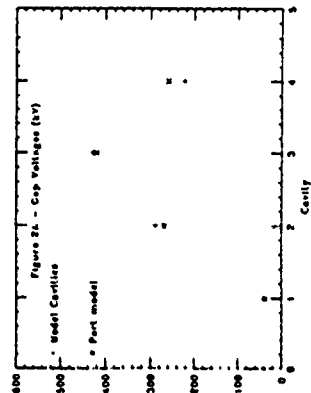


Figure 2. Comparison between full cavity simulation and the impedance matrix port model for the case shown in Figure 1. Figure 2A shows the cavity voltages for each cell and Figure 2B shows the cavity phases.



SST - A Single-Slice Particle Simulation Code*

Jonathan Krall, Glenn Joyce and Steve Slinker

Beam Physics Branch
 Plasma Physics Division
 Naval Research Laboratory
 Washington, DC 20375-5000

Transport studies of high-current relativistic electron beams in advanced induction accelerator configurations has required the development of sophisticated particle simulation codes. Recently, we have been interested in magnetic transport of such a beam within a metallic pipe. We have found that, to a very reasonable approximation, these simulations can be performed by following the transport of a single "slice" of the intense beam in (r, θ, z) coordinates under the approximations that $\partial/\partial z = 0$ and $z = ct$, where z is the axial coordinate and c is the speed of light. Thus we have developed the SST (single-slice transport) code which was derived from the family of beam/plasma simulation codes which includes the ELBA¹ and FRIEZR² codes.

Previously, we developed the FRIEZR² code to simulate axisymmetric beam propagation in the Ion Focus Regime (IFR). Numerous studies were undertaken using that code, including simulations of IFR propagation in the Advanced Test Accelerator.³ Interest in the study of three-dimensional beam propagation and transport phenomena such as the electron-hose instability⁴ of a beam in a diffuse plasma and the electromagnetic three-wave instability⁵ of a beam in strong-focusing magnetic fields inspired the development of a three-dimensional code, ELBA. The ELBA code was based largely on FRIEZR with the addition of three-dimensional electromagnetic fields. It was formulated in the (r, θ, z, ct) coordinate system, has fully relativistic beam and plasma particles, and is fully electromagnetic. In

addition, ELBA included the capability of simulating the transport of particle beams around bends in the limit of a large radius of curvature. The curved-geometry formulation was based on a mapping scheme developed by Alex Friedman.⁶ The mapping algorithm does not depend on the radius of curvature, but the electromagnetic fields include an expansion to first order in the inverse aspect-ratio of the the simulation pipe radius to the radius of curvature.

The SST code is effectively a version of the ELBA code with the constraints $\partial/\partial z = 0$ and $z = ct$. This provides a major simplification in that the 1000-4000 particles of the "slice" constitute a small memory requirement, making the memory management and particle sorting that are done by ELBA unnecessary. Whereas the ELBA code was written for the Cray series of computers, the SST code was designed to run on a workstation. Thus the parallelized (and confusing) laydown scheme of that code has been replaced with a more standard loop over the particles. To avoid sorting, we keep all particles in memory, even those that have hit the outer radial boundary. These particles are simply turned "off" and are bypassed via IF/THEN statements. We further simplified SST by considering only one species of particles (i.e. there is no plasma). While a typical run of the ELBA code consumed 10 minutes on a Cray X-MP computer, the equivalent SST simulation uses 5 minutes on a SiliconGraphics Personal Iris workstation.

The SST code employs an (r, θ) coordinate system with a constant Δr , constant $\Delta \theta$ grid (nonlinear Δr grids have also been used). The number of grid points in θ is constrained to be a power of two (typically 8 or 16) by the field solver. The assignment of the axial coordinate, z , is required to make reference to external magnetic fields. The approximation $z = ct$ is reasonable for relativistic electron beams. If mildly relativistic

particles such as ions were implemented, $z = v_b t$ could equivalently be used, where v_b is the average axial velocity of the particles.

The SST code was written such that it retained the ability of the ELBA code to simulate transport of the beam through a curved pipe. Under the $\partial/\partial z = 0$ approximation, we simulate either an infinite beam in an infinite pipe (straight section) or a current ring in a toroidal chamber (curved section). Because there are no axial variations, this transition is made instantaneously but without adverse effect.

At each SST time step we calculate the electromagnetic fields for the slice, add the external fields, push the particles and calculate the new charge and current densities. The push is carried out in rectangular coordinates using standard methods. For diagnostic purposes, the axial momentum, while not used, is updated at each step. The particles are typically initialized to t in equilibrium with the external magnetic field configuration at $z = 0$, as has been done with some versions of the ELBA code.⁷ Here the magnetic field configuration consists of a solenoidal and/or a stellerator (twisted quadrupole) field. Note that the initialization of reference 7 is an equilibria only in the context of an envelope model⁸ and is valid for $v/v \ll 1$, where $v = I_b[kA]/17$ and v is the usual relativistic factor for the beam.

The electromagnetic fields are obtained from a representation of the scalar and vector potentials. The equations are solved in much the same manner as in the ELBA code, using the gauge $\nabla_j \cdot A_j = 0$, but with $\partial/\partial z = 0$. Here, A_1 is the component of the vector potential perpendicular to the z axis. The potentials are determined mode by mode in a Fourier transformed space, where the Fourier transform is taken in the azimuthal direction. For

initial conditions, we choose a time independent solution for the electromagnetic fields appropriate to a beam in a vacuum.

The SST code has proven to be a useful tool in a number of studies of relativistic beam transport in external magnetic fields. We have restricted the boundary conditions to be those of a perfectly conducting cylinder so that the electromagnetic fields can be solved quickly. As a result, the code is fast enough to be useful in a workstation environment. With its ability to take the beam from a straight section of beam pipe into a curve and then out again, this code is a unique tool for induction accelerator design studies.

- [1] G. Joyce, J. Krall and S. Slinker, Proc. of the Conf. on Computer Codes in the Linear Accel. Community, LANL Report LA-11857-C, 99 (1990).
- [2] J. Krall, K. Nguyen, and G. Joyce, Phys. Fluids B 1, 2099 (1989).
- [3] R. F. Hubbard, G. Joyce, S. P. Slinker, J. Krall and M. Lampe, Bull. Am. Phys. Soc. 33, 1950 (1988).
- [4] D.H. Whitlum, V. M. Sharp, S. Yu, M. Lampe and G. Joyce to appear in Phys. Rev. Lett. (1991).
- [5] J. Krall, C. M. Tang, G. Joyce and P. Sprangle, Phys. Fluids B 3, 204 (1991); T. P. Hughes and B. B. Godfrey, Phys. Fluids 29, 1690 (1986).
- [6] A. Friedman, "Methods for PIC Simulation of Bent Beams in 3d and 2d", paper PB8-10, 13th Conference on Numerical Simulation of Plasmas, (1989).
- [7] J. Krall and G. Joyce, Proc. of the Conf. on Computer Codes in the Linear Accel. Community, LANL Report LA-11857-C, 341 (1990).
- [8] D. Chernin, Particle Accelerators 24, 29 (1988).

*Supported by DARPA, ARPA Order Number 7781, monitored by NSVC.

ENVELOPE MODEL OF A RECIRCULATING HEAVY-ION ACCELERATOR*

W. M. Sharp, J. J. Barnard, and S. S. Yu

Lawrence Livermore National Laboratory
Livermore, California 94550

A fast-running envelope code has been developed to carry out the detailed lattice design for the Lawrence Livermore National Laboratory (LLNL) recirculating heavy-ion accelerator.¹ The model combines an envelope description of the beam transverse dynamics with a fluid-like treatment of axial dynamics. Appropriate terms are included to account for the effects of image forces, beam emittance, and space-charge in the limit of paraxial motion, and the beam is focused and accelerated by a user-specified lattice of time-varying quadrupoles, bending magnets, and accelerating modules. Since the recirculator is expected to require bends out of the nominal accelerator plane to allow beam injection and extraction, the model allows a fairly general three-dimensional (3-D) lattice specification, and the focus of this paper is the generalization of the previous co-planar model to include this 3-D feature.

The set of envelope equations used here to model the beam transverse dynamics is a generalization of that formulated first by Kapchinskij and Vladimirov and later adapted by Lee, Close, and Smith.² The beam transverse distribution function is assumed to be uniform and elliptical in each phase-space plane, and in the absence of an axial magnetic field the coordinate-space ellipse taken to be unskewed. In fact, terms arising from the bent coordinate system introduce a weak coupling between the envelope radii that could lead the skewing of the ellipse, but these terms are small enough to be neglected. Except for image forces, the treatment is first order in the ratio of the beam radii and centroid displacements to the beam-pipe radius. This approximation makes it appropriate to represent bend magnets and quadrupoles by idealized expressions and to neglect higher-order multipole fields. In the present version of the code, only single-function magnets are used, and a simple analytic model may optionally be used to represent the axial fringe fields. Although the ion mass is multiplied where it appears by the Lorentz factor γ , derivatives of γ are small enough to be dropped from the ion-motion equations. Also, we assume a circular beam pipe of infinite conductivity and radius R .

* This work was performed under the auspices of the US Department of Energy by Lawrence Livermore National Laboratory under W-7405-ENG-48.

¹ S. S. Yu, J. J. Barnard, G. J. Caporaso, A. Friedman, D. W. Hewitt, H. Kirbie, M. A. Newton, V. K. Neil, A. C. Paul, L. L. Reginato, W. M. Sharp, R. O. Bangert, C. G. Fong, D. L. Judd, T. F. Godlove, "Constraints on Recirculating Induction-Driver Designs," presented at the 1990 International Symposium of Heavy-Ion Inertial Fusion, Monterey, CA (to be published).

² E. P. Lee, E. Close, and L. Smith, "Space Charge Effects in a Bending Magnet System," in *Proceedings of the 1987 IEEE Part. Accel. Conf.*, E. R. Lindstrom and L. S. Taylor, eds (IEEE, New Jersey, 1987), p. 1126.

With these assumptions, the coupled envelope equations for the coordinate-space radii may be written as follows:

$$\frac{d^2 a}{ds^2} + \frac{1}{\beta} \frac{d\beta}{ds} \frac{da}{ds} = \pm \frac{B'}{[B\rho]} a + \frac{c^2}{a+b} + \frac{2K}{a+b} + \frac{f_0 K}{R^2} a + \frac{f_1 K}{R^2} b \quad (1a)$$

$$\frac{d^2 b}{ds^2} + \frac{1}{\beta} \frac{d\beta}{ds} \frac{db}{ds} = \mp \frac{B'}{[B\rho]} b + \frac{c^2}{a+b} + \frac{2K}{a+b} - \frac{f_0 K}{R^2} b + \frac{f_1 K}{R^2} a. \quad (1b)$$

Here, the coordinate s is distance around the axis of the lattice, and x and y are respectively the spatial coordinates in and perpendicular to the plane of the lattice, with a and b being the beam coordinate-space radii in the x and y directions. The quadrupole-field transverse gradient is denoted by B' , with the sign being determined by the magnet orientation. The "magnetic rigidity" in the quadrupole-focusing terms is given in SI units by $[B\rho] = \beta\gamma Mc/qc$, where M and q are the ion mass and charge state, β is the axial velocity scaled by the speed of light c , and γMc^2 is the total energy of beam ions. The prevertex K in the space-charge and image-force terms is defined as

$$K = \frac{1}{4\pi\epsilon_0} \frac{2geI_b}{(\beta\gamma)^2 Mc^2}, \quad (2)$$

where I_b is the beam current in Amperes, and ϵ_0 is the free-space permittivity. The beam transverse temperature is accounted for in Eq. (1) by the terms proportional to the unnormalized emittances ϵ_x and ϵ_y , which are calculated here by assuming that the normalized-emittance components $\beta\gamma\epsilon_x$ and $\beta\gamma\epsilon_y$ are constant. Following the method in Ref. 3, the image-force terms in Eq. (1) were derived by assuming that the centroid of the elliptical beam is displaced a distance $(X^2 + Y^2)^{1/2} \ll R$ from the axis of a straight beam pipe, where X and Y are the beam centroid coordinates. With this assumption, the coefficients f_0 and f_1 have the forms

$$f_0(a, b, X, Y) = \frac{a^2 - b^2}{4R^2} + \frac{X^2 - Y^2}{R^2} + \frac{3}{2} \frac{X^2 + Y^2}{R^2} \left(\frac{a^2 - b^2}{R^2} \right) + \frac{3}{8} \frac{X^2 - Y^2}{R^2} \left(\frac{a^2 - b^2}{R^2} \right)^2 \quad (3)$$

$$f_1(a, b, X, Y) = \frac{2XY}{R^2} \left[1 - \frac{3}{8} \left(\frac{a^2 - b^2}{R^2} \right)^2 \right]. \quad (4)$$

The presumption of a straight beam pipe substantially simplifies the algebra and is appropriate when the bend radius ρ is much larger than R . The $d\beta/ds$ term in Eq. (1) arises from changing variables from t to s , and, as discussed below, an approximate expression for the derivative is obtained directly from one of the equations for longitudinal motion

Equation for the lattice-plane centroid coordinates X and Y are obtained from a distribution average of the single-particle motion equations. For a dipole field strength B_d that is rotated by an angle α from the positive y direction, we find

$$\frac{d^2 X}{ds^2} + \frac{1}{\beta} \frac{d\beta}{ds} \frac{dX}{ds} = \pm \frac{B'}{[B\rho]} X + \left[\frac{1}{\rho} - \left(1 + \frac{X \cos \alpha + Y \sin \alpha}{\rho} \right) \frac{B_d}{[B\rho]} \right] \cos \alpha$$

$$+ \frac{g+K}{R^2} X - \frac{hK}{R^2} Y - \left[\frac{X \cos \alpha + Y \sin \alpha}{\rho^2} \right] \cos \alpha, \quad (5a)$$

$$\frac{d^2 Y}{ds^2} + \frac{1}{\beta} \frac{d\beta}{ds} \frac{dY}{ds} = \mp \left[\frac{D'}{B\rho} Y + \left(\frac{1}{\rho} - \left(1 + \frac{X \cos \alpha + Y \sin \alpha}{\rho^2} \right) \frac{B_4}{B\rho} \right) \frac{\sin \alpha}{\rho} \right] + \frac{g-K}{R^2} Y - \frac{hK}{R^2} X - \left[\frac{X \cos \alpha + Y \sin \alpha}{\rho^2} \right] \sin \alpha, \quad (5b)$$

where ρ is the bend radius, and the image-force coefficients g and h are given in the straight-pipe paraxial limit by

$$g \pm (a, b, X, Y) = 1 \pm \frac{a^2 - b^2}{4R^2} \pm \frac{(X^2 - Y^2)}{R^2} \pm \frac{3}{4} \frac{(X^2 + Y^2)}{R^2} \left(\frac{a^2 - b^2}{R^2} \right) \pm \frac{1}{8} \frac{(X^2 - Y^2)}{R^2} \left(\frac{a^2 - b^2}{R^2} \right)^2, \quad (6a)$$

$$h(a, b, X, Y) = \frac{2XY}{R^2} \left[1 - \frac{1}{8} \left(\frac{a^2 - b^2}{R^2} \right)^2 \right]. \quad (6b)$$

In Eq. (5), we have chosen to use B_y explicitly in the momentum-difference term so that a bend field may be applied for extraction in a straight section of the lattice. When the bend is rotated out of the nominal lattice plane by some angle ϕ , the vector components in Eqs. (5) - (6) must first be represented in terms of their coordinates in the unrotated frame, and the equations for X and Y in that frame are obtained by appropriate linear combinations of the bend-plane equations.

To model axial dynamics, we treat slices of the beam as Lagrangian fluid elements characterized by an axial velocity βc and the time τ that the slice arrives at an axial location s . This approach implicitly assumes that the beam has a negligible longitudinal temperature and that the slices remain approximately collinear. If the slice boundaries are presumed to remain perpendicular to the beam-pipe axis, then the equation for τ is found from orbit kinematics to be

$$\frac{d\tau}{ds} = \frac{1}{\beta c} \left(1 + \frac{X}{\rho} \right), \quad (7)$$

where we have again assumed paraxial motion. An approximate β equation is obtained by retaining only the electrostatic force in the single-particle motion equations and averaging the axial component over the beam elliptical cross-section:

$$\frac{d\beta}{ds} = \frac{qc}{\beta M c^2} \left(1 + \frac{X}{\rho} \right) (E_{ext} + E_{ec}). \quad (8)$$

Here, the average external electric field E_{ext} is approximated only by the voltage across accelerating modules divided by the gap length. The axial electric field induced by the change in bend-magnet field strength is neglected here because the field has a peak strength

that is less than 0.1% of the average accelerating gradient and furthermore has a null on axis. The space-charge field is approximated by

$$E_{ec} \approx \frac{1}{4\pi\epsilon_0} \left[\ln \left(\frac{2R^2}{a^2 + b^2} \right) + \frac{1}{2} \right] \left[\frac{\partial}{\partial r} \left(\frac{\lambda}{\beta c} \right) + \frac{\lambda}{\beta} \frac{d\beta}{ds} \right], \quad (9)$$

where the line-charge density λ for a slice containing charge ΔQ is estimated by

$$\lambda = \frac{\Delta Q}{\beta c \Delta \tau}. \quad (10)$$

In deriving the space-charge field, the radial electrostatic field is assumed to vary over a much shorter scale length than E_{ec} , and the continuity equation is used to convert derivatives with respect to s into τ derivatives. When Eq. (9) is substituted into the β equation Eq. (8), the resulting equation is trivially rearranged to give an equation for $d\beta/ds$ in terms of E_{ext} and the time derivative of λ/β .

The equations Eqs. (1) - (10) are recast in the code as a set of ten first-order equations and are integrated by a conventional fourth-order Runge-Kutta method. A constant step size in s is used except near the boundaries of lattice elements, where the step is chosen to land on each boundary. The results are found to be insensitive to the choice of step size so long as there are 10 or more integration steps per lattice element. To initialize the equations in equilibrium, we integrate the equations over the first full lattice period and use a vector form of Newton's method to adjust the initial values of the beam radii and their derivatives in each slice until they equal the corresponding final values.

An important aspect of the code is the lattice specification. The user may specify an arbitrary number of distinct lattice elements, specifying the such appropriate properties as length, aperture, strength, bend angle, and the rotation angle in the plane transverse to the beam direction. At present, the element types allowed in the code are drifts, accelerating gaps, sector bend magnets, and quadrupoles, but solenoids and higher-order multipoles may be easily added. Each lattice element is given a name by the user, and lattice sections may be defined by listing names of previously defined elements and subsections along with the number of times the listed items are repeated. The final such grouping of subsections is treated as the complete lattice, and the specification is checked to determine whether the lattice is a closed loop.

To facilitate code use, the code has a lattice "self-design" option. The main assumption used to modify the lattice parameters is that energy is gained linearly in distance s around the ring. With this assumption, the time-averaged voltage across accelerating gaps and the bend-field strengths and time derivatives may be set. Also, some compensation for longitudinal space charge is possible by imposing an appropriate time variation on the accelerating voltage. In the code, this voltage is allowed to vary up to quadratic order with time, and the voltage-correction terms are calculated from a four-term Legendre-series representation of the initial beam space charge.

ELBA - A Three Dimensional Particle Simulation Code for High Current Beams

S. Slinker, G. Joyce, J. Krall, and R.F. Hubbard

Beam Physics Branch
Plasma Physics Division
Naval Research Laboratory
Washington, DC 20475-5000

INTRODUCTION

The study of the interaction of intense relativistic electron beams with plasmas and with external magnetic field configurations has required the development of sophisticated particle simulation codes. Proper modeling of the self-fields of these beams is important. We developed the FRIEZR code to look at axisymmetric problems of beam propagation in the ion-focused regime (IFR). A number of investigations were made using the models integrated into that code. We are now progressing to the study of three dimensional phenomena associated with beam propagation through diffuse plasmas and with beam transport in compact accelerators. We have developed a three dimensional code, ELBA, to investigate the physics of some of these phenomena. The ELBA code is based largely on FRIEZR with the addition of three dimensional electromagnetic fields. It is formulated in the (r, θ, z) coordinate system, has fully relativistic beam and plasma particles, and is fully v-coupled. In addition, we have included the capability of simulating the transport of particle beams around bends in the limit of a large radius of curvature. The latter formulation is based on a mapping scheme developed by Friedman.^{2,3} The mapping algorithm is valid for any radius of curvature, but the electromagnetic fields include an expansion to first order in the inverse aspect ratio of the simulation waveguide radius to the radius of curvature.

GENERAL ISSUES

Because of the nature of high energy electrons, a number of conventions have been adopted for their study which are not usually found in particle simulation codes. It is convenient to replace the space and time variables x and t by $\zeta = ct - z$ and $\tau = ct$. Relativistic beam particles, in terms of these variables, exhibit very slow motion in the axial, ζ , direction. A beam particle moving exactly at the speed of light would have a constant value of ζ . The actual motion will result in a slow increase in ζ position of the particle depending on its energy and perpendicular velocity. The plasma particles on the other hand appear to be streaming by the beam at roughly the speed of light. We are usually interested primarily in the beam dynamics so the ζ, τ coordinates are generally well suited for our use. A second consequence of the ζ, τ coordinates concerns the boundary conditions for the electromagnetic fields. No information can flow forward in these variables so the propagation of all electromagnetic waves is toward larger ζ . As a result, we can predetermine the values of the electromagnetic fields at the front end of the simulation system and they will remain unchanged by the beam.

The fact that we do not have to treat information moving in the forward direction allows us to handle particle data in a particularly simple way. The

simulation region is divided axially into a number of "slices" with the slice length corresponding to the grid size along the ζ axis. The point $\zeta=0$ is generally associated with initial position of the beam head and $\zeta=\zeta_{\max}$ is associated with the beam tail. At any given time each particle, whether it is a beam or plasma particle, is associated with a particular slice. As time progresses, the particle either stays in that slice or moves to the next slice at larger ζ . A particle at any given slice cannot be influenced by a particle in another slice which is nearer the tail beam. As a result, not all particles need to be in the computer simultaneously.

At each time step we can begin with the first slice, read the particle variables for this slice (with some associated field, charge, and current density quantities), calculate the electromagnetic fields for the slice, push the particles, calculate the charge and current densities, store the particle density, and field information on disk and begin with the next slice working our way from the front to the back of the beam. As particles move from slice to slice, they are marked as having moved to next slice and stored appropriately. The marking process requires that we must do a particle sort at each time step and at each slice. The sort is into two regions: the particles which remain in the slice, and those which move to the next slice. At the time that we do the sort, we also handle particles which have crossed the radial simulation boundary. Normally, we excise these particles from the simulation. As a result of this buffering system, we can simulate beams with an almost unlimited number of particles although we pay a price in I/O. A single time step represents the processing of all the slices in the simulation region from the beam head to the beam tail. No particle can move to a slice nearer the beam head since the slice "velocity" is c . The beam particles slowly move from slice to slice while the plasma particles change slices frequently. In order to make the bookkeeping simple, we choose the time step to be smaller than $\Delta t/2c$. This is a constraint is also imposed on us by the field solver.

The particle push is the standard leapfrog algorithm for the Lorentz force equations and is fully relativistic for the beam and plasma (both plasma ions and electrons can be treated). Besides the three (Cartesian) positions and three momenta, each particle carries its own charge weighting. The simulation particles are stored in two categories, electrons (both beam and plasma) and ions, so that the difference in mass can be accounted for in the push.

ELECTROMAGNETIC FIELDS

We solve the electromagnetic fields from a representation of the scalar and vector potentials using cylindrical coordinates. The equations are solved in much the same manner as FRIEZR using the Lee gauge⁴ $\nabla \cdot \mathbf{A} = 0$. Here, \mathbf{A} is the component of the vector potential perpendicular to the ζ axis. The Lee gauge works well for problems in which the beam is nearly axisymmetric and dominates the field production. In addition, some versions of the code use the Lorentz gauge which works better for studying propagation of laser beams.

In the variables (ζ, r) , the differential operators become $\frac{1}{c} \frac{\partial}{\partial \zeta} + \frac{\partial}{\partial r} + \frac{\partial}{\partial \zeta}$ and Maxwell's equations in the Lee gauge can be written

$$\nabla^2 \phi + \frac{\partial^2 \phi}{\partial \zeta^2} = -4\pi\rho \quad (1)$$

$$\nabla^2 a - \frac{\partial}{\partial \zeta} \left(\frac{\partial a}{\partial r} + \frac{\partial}{\partial \zeta} \right) a = -4\pi(j_z/c - \rho) + \frac{\partial^2 \phi}{\partial \zeta^2} \quad (2)$$

$$\nabla^2 \lambda_1 - \frac{\partial}{\partial \zeta} \left(\frac{\partial \lambda_1}{\partial r} + 2 \frac{\partial \lambda_1}{\partial \zeta} \right) \lambda_1 = -\frac{4\pi}{c} j_1 - \nabla_1 \left(\frac{\partial a}{\partial \zeta} - \frac{\partial \phi}{\partial r} \right) \quad (3)$$

$$E_z = - \left(\frac{\partial}{\partial r} + \frac{\partial}{\partial \zeta} \right) a - \frac{\partial \phi}{\partial \zeta} \quad (4)$$

where $a = A_z - \phi$.

We solve Eqs. (1) through (3) with the gauge condition made by mode in a Fourier transformed space where the Fourier transform is taken in the azimuthal direction. Because of our choice of variables, the hyperbolic equations admit only solutions which are stationary or are moving toward larger ζ . We have chosen to difference the equations explicitly in the ζ direction and implicitly in the radial direction. The outer radial boundary is assumed to be a perfect conductor. The implicit differencing was chosen to avoid a Courant condition in the radial direction where the spatial grid size is generally smaller than axially. There is a Courant condition in the axial direction which turns out to be $\delta r \leq \delta \zeta/2$.

Once the potentials are found, the fields needed to push the particles are calculated. At this time external fields are added. Beam transport in ideal solenoids and twisted quadrupoles has been studied. The code can also use experimentally measured on-axis axial magnetic field data, if the Taylor series expansion for the off-axis values is valid. For the potentials in the bends, Maxwell's equations are expanded in the ratio of the simulation tube radius to the radius of curvature of the bend, and only first order terms in this ratio are retained.

We have chosen a radial grid which is non-linear in the r coordinate. Since the beam particles normally lie near the axis of the system, it is convenient to have the grids more closely spaced here than near the walls. An evenly spaced grid in the coordinate $u = \ln(r + r_0)$ has that property. Near the axis $u = r^2$, which approximates a constant current density grid, while for large r the grid spacing increases exponentially.

PARTICLE LAYDOWN

When ELBA is used to model situations in which the plasma dynamics are important, it is subject to a buildup of short wavelength noise in the ζ direction. The dominant noise appears to be at about the Nyquist wavelength. We have used a standard method to suppress the noise which is to average the current gridpoint with the two axially adjacent gridpoints at a ratio of 1-2-1. This is equivalent to applying a filter function of the form $0.5(1 + \cos k_0 \zeta)$ to the Fourier spectra of the densities where k_0 is the axial wavenumber. This has the effect of zeroing out fluctuations at the shortest wavelengths of the system.

When applying the laydown algorithm to the particles, there are a number of quantities to which each particle can contribute. For ELBA the particles are linearly interpolated in ζ , r and θ for calculation of the charge density and the three current densities. Thus, each particle contributes to 32 different quantities. By looping over the 32 density components, it is possible to vectorize the particle laydown scheme. There is a tradeoff between increased memory usage and computational efficiency in using this method. In addition, the logic of the code becomes much more difficult to understand when using this form of the particle laydown.

INITIAL CONDITIONS

Although the choice of the (ζ, r) coordinate system simplifies the boundary conditions at $\zeta=0$, some decisions must be made about the laboratory $z=0$ and the initial conditions. The simplest assumption is that it is not important where the beam is initially. In this case, the beam charge can be turned on adiabatically, and the long term behavior of the system can be studied. Usually, however, we want to simulate the beam entering a region of interest from some other region and we must decide how to treat $(z=0, r=0)$. After a number of trials, we have implemented the following initialization steps. At $t=0$, we assume that the head of the beam is at $z=0$ and that the rest of the beam is out of the region of interest. For initial conditions, we choose a time independent solution for the electromagnetic fields appropriate to a beam in a vacuum. The plasma is chosen to be charge neutral and cold. As the beam enters the simulation region ($z>0$), we push the beam particles, but for the portion of the beam remaining outside the simulation region, we freeze motion in the radial direction and allow the beam to free stream in the ζ direction. In order to avoid discontinuities at the $z=0$ boundary, we allow the plasma to react to the fields in both regions of the simulation. This scenario provides a reasonable model for the entrance of the beam into a transport system.

CONCLUSIONS

The ELBA code has proven to be a useful tool in a number of studies of relativistic beam transport and propagation. The coordinate system limits its applicability to beams which are moving at roughly the speed of light. We have restricted the boundary conditions to be those of a perfectly conducting cylinder so that the electromagnetic fields can be solved quickly. We have usually opted for speed and simplicity over generality.

1. J. Krall, K. Nguyen, and G. Joyce, Phys. Fluids B 1, 2099 (1989).
2. A. Friedman, "Methods for PIC Simulation of Bent Beams in 3d and 2d," paper PHB-10, 13th Conference on Numerical Simulation of Plasmas. (1989).
3. A. Friedman, et al., "WARP A 3D (+) PIC Code for HIF Simulation," Proceedings of the Conference on Computer Codes in the Linear Accelerator Community, Los Alamos National Laboratory, LANL Report 11857-C, pp. 119-136, (1990).
4. E.P. Lee, "The New Field Equations," UCID 17826, Lawrence Livermore Laboratory, (1976).

Work supported by DARPA, ARPA Order 7781, monitored by NSWC.

Nonlinear Simulation of Tokamak Turbulence: Field-Line-Following Coordinates and Global Gyrokinetics

A. M. Dimits

Lawrence Livermore National Laboratory
P.O. Box 5511, Livermore CA 94550

I Nonlinear Simulation using Field-Line-Following Coordinates

Many instabilities of interest in tokamak plasmas, for example ion temperature gradient driven (ITG) modes, trapped-electron modes, and pressure-driven MHD modes have a high- n ballooning structure ($k_{\parallel} \ll k_{\perp}$). This structure can be exploited to greatly reduce the computer time required for nonlinear simulations, by using a set of coordinates in which one of the coordinate directions lies approximately (not necessarily exactly) along the magnetic field. This permits the use of a coarse grid in that coordinate direction. If, for example, V , θ , and ζ are radial, poloidal and toroidal straight-field-line coordinates, then suitable coordinate choices are (V, θ, β) , where $\beta \equiv \zeta - \tilde{q}(V)\theta$, with a coarse grid in $\theta[1]$, or (V, X, ζ) , where $X \equiv \theta - \zeta/\tilde{q}(V)$, with a coarse grid in ζ . Here $\tilde{q}(V) \approx q(V)$, where $q(V)$ is the magnetic safety factor. The main complication that arises is in the implementation of the correct periodicity conditions [$\psi(V, \theta, \beta) = \psi(V, \theta, \beta + 1) = \psi(V, \theta + 1, \beta - \tilde{q})$ in the first case, and $\psi(V, X, \zeta) = \psi(V, X + 1, \zeta) = \psi(V, X - 1/\tilde{q}, \zeta + 1)$ in the second case]. A simple solution to this problem has been obtained[2], based on exact offset-periodic meshing of the grid lines at the opposite boundaries of the box (in the θ - β or X - ζ planes). The method is applicable to finite-difference and spectral fluid simulations, as well as to (gyrokinetic) particle simulations.

Consider the implementation in the (V, X, ζ) coordinates. If there are N_X grid points in the X direction then the values of the rotational transform of the new coordinate system available when the $X = \text{const.}$ grid lines are meshed at $\zeta = 0$ and $\zeta = 1$ are $i \equiv 1/\tilde{q} = m/N_X$, where m is an integer. For a given profile of the magnetic rotational transform $i(V) \equiv q^{-1}(V)$, choose integers $m(V)$ at each radial grid location V so as to minimize $i(V) - i(V)$. The parallel derivative is given by $\nabla_{\parallel} = \partial/\partial\zeta + (i - i)\partial/\partial X$. For highly elongated structures, we have $\partial/\partial\theta \gg \nabla_{\parallel}$, so that if $i - i = O(1)$ (as is generally the case for $i = 0$), as many grid points are needed in the ζ direction as in the θ direction. Two large derivatives must approximately cancel to yield a small derivative. When $m(V)$ is chosen so that $(i - i)\partial/\partial X \approx \nabla_{\parallel}$, on the

other hand, the ζ derivative need only resolve the parallel variation of the structure, so that far fewer grid points are now needed in the ζ direction. The reduction in the number of grid points that can be attained is proportional to the elongation of the structures. For ITG turbulence, for example, a reduction of order 10^{-2} is to be expected.

The implementation of this method is quite straightforward, and quite general q profiles can be well fitted, given the values of N_X that are necessary for adequate resolution in the X direction. The form of $\mathbf{E} \times \mathbf{B}$ nonlinear fluid advection terms does not change under the $\theta \rightarrow X$ transformation. A continuous-in- ζ discretization of the radial derivative, needed to handle finite-gyroradius effects, is given by

$$\begin{aligned} \frac{\partial \psi}{\partial V} \Big|_0 &= \frac{\partial \psi}{\partial V} \Big|_X - \zeta'(V) \frac{\partial \psi}{\partial X} \\ &\rightarrow \frac{1}{2\Delta V} \{ \psi(i+1, j, \zeta) - \psi(i-1, j, \zeta) + \zeta(\psi(i+1, j + i(i+1) - i(i), \zeta) \\ &\quad - \psi(i+1, j, \zeta)) + \zeta(\psi(i-1, j, \zeta) - \psi(i-1, j + i(i-1) - i(i), \zeta)) \}, \end{aligned}$$

where ΔV is the radial grid spacing, and i and j are respectively the V and X grid indices. Higher-order discretizations follow trivially, being combinations of the above form. Fast Fourier transforms can be used to obtain the discrete Fourier transform in the X - ζ plane, using the result that $\sum_{\ell=1}^{MN_X} \psi(k_X, \ell) \exp(2\pi i k_X \ell / (M N_X))$ is given by

$$\psi(k_X, j) = M \sum_{\ell=1}^{N_X} \exp(2\pi i j \ell / N_X) \exp(-2\pi i k_X \ell / (N_X N_X)) \psi(k_X, \ell),$$

where N_X is the number of points in the ζ direction and M is the number of passages of a $X = \text{const.}$ grid line before it loops back on itself, and $k_X = M(j - k_X i / N_X)$. Fast Fourier transforms in the radial direction may also be useful if $i(V)$ is either constant or is linear with a sufficiently large gradient.

The method has been implemented in a three-dimensional finite-difference nonlinear fluid code and its usefulness for ion-temperature-gradient-driven turbulence has been demonstrated[2]. For fluid or Vlasov-fluid type simulations, the reduction in computer time needed per timestep decreases in proportion to or more rapidly than the decrease in the number of grid cells needed. Also, in many cases, the coarse parallel grid automatically filters out physically irrelevant but numerically problematic high-frequency modes, permitting much longer timesteps for explicit timestepping. Results for sheared magnetic fields will be presented. Work on the implementation of the method in a three-dimensional partially-linearized[3] ("sf") gyrokinetic particle

code is presently underway, and results will be presented. It is anticipated that the use of these coordinates in a particle code will reduce the particle number requirement through the automatic filtering of noisy short-parallel-wavelength modes, and also result in direct reductions in the computer time required for the interpolation of the charge (and current) densities and for the solution of the field equations. These grid-cell-number-scaling savings become important when the reduction in the particle number allowed by the filtering is taken into account. Results will be shown that address the relative importance of these contributions to improved performance of the code.

II. Global Gyrokinetic Simulation

The gyrokinetic equations have recently been rederived in a generalized ordering that permits the simultaneous existence of fluctuations at all radial scale lengths ranging from the gyroradius scale to the profile gradient scales[4]. While the appearance of the resulting equations is formally very similar to that of previously derived equations[5, 6], the gyrokinetic Poisson equation (or Maxwell's equations in the electromagnetic case)

$$\nabla^2 \phi(\mathbf{z}, t) = -4\pi e \left\{ \int [F_i(\mathbf{z}, t) + e \partial F_i / \partial \mu] \delta(\mathbf{X} - \mathbf{z} + \rho) d^3 \mathbf{Z} - n_e \right\} \quad (1)$$

(where ϕ is the potential, $\phi \equiv \phi(\mathbf{z}, t) - (1/2\pi) \oint d\alpha \phi(\mathbf{X} + \rho(\alpha), t)$, $\mathbf{Z} \equiv (\mathbf{X}, \mu, \theta, v_\parallel)$ are the gyrocenter phase space coordinates, with μ the magnetic moment, F_i is the gyroaveraged distribution function, and e is the generalized gyroviscous expansion parameter), must be reinterpreted. If long-wavelength fluctuations are present, then $\partial F_i / \partial \mu$ can no longer be taken to be the result for the background distribution, but must be calculated from the particle data. The second term on the RHS of Eq. (1) can be integrated by parts (neglecting higher-order terms), the gyroangle integrations replaced by averages over fit numbers of points[7], and the resulting densities given a weighted assignment to the grid points to produce a term of the form

$$[L\phi](\mathbf{z}_{(i,k)}) = N_\theta^{-2} \sum_i \rho_i^{-2} \sum_{(k)} W[X_i + \rho_i(\theta_i) - \mathbf{z}_{(i,k)}] \times [\rho_i(\theta_i) - \rho_i(\theta_i)] \cdot \nabla \phi(\mathbf{z}_{(i,k)} - \rho_i(\theta_i) + \rho_i(\theta_i')).$$

where $\mathbf{z}_{(i,k)}$ is the physical position of grid point (j, k) in the radial-poloidal plane, $N_\theta^{-1} \sum_{(k)}$ represents an average over N_θ values of θ_i , i is the particle index, W is the grid charge assignment function, $\rho_i(\theta)$ is the gyroradius vector for a particle i with gyroradius ρ_i and gyroangle θ . The details of the operator L depend on the

order of the discretization of the derivatives of ϕ . Investigations of the feasibility of solving the resulting matrix equation by relaxation or other methods are underway. If particles with gyroradii ρ_i up to three times the grid spacing are kept, then the equation involves a $N_\theta \times N_\theta$ matrix with of order 10^3 nonzero elements per row. The computational expense of accumulating these elements from the particle data at each time step may be prohibitive. Also, although the resulting matrix is likely to be definite under most practical circumstances, this may not always be the case. A feasible alternative is to use a finite-gyroradius-expanded form of the gyrokinetic Poisson equation. This will result in a code that is valid only for fluctuations longer than the gyroradius, but otherwise fully kinetic. The resulting matrix equation must still be calculated from the particle data but now has few nonzero elements per row (e.g., five for 3-point centered differencing). Progress will be reported on work underway to modify a two-dimensional gyrokinetic code to implement this scheme.

Acknowledgments

The author gratefully acknowledges useful discussions with T. M. Antonsen Jr., S. C. Cowley, J. F. Drake, R. H. Cohen, and L. LoDestro. This work was performed by LLNL for USDoE under contract W-7405-ENG-48.

References

- [1] A. H. Glasser, private communication.
- [2] A. M. Dimits, International Sherwood Fusion Conference, (STI Optronics, Inc. Seattle, WA, paper 1C18).
- [3] A. M. Dimits, PhD thesis, (Princeton University, Princeton, NJ, Feb 1988)
- [4] A. M. Dimits, L. LoDestro, and D. H. E. Dubin, LLNL report UCRL-JC-106136 (1991, submitted to Phys. Fluids B).
- [5] W. W. Lee, Phys. Fluids 26, 556 (1983).
- [6] D. H. E. Dubin, J. A. Krommes, C. Oßerman, and W. W. Lee, Phys. Fluids 26, 3524 (1983).
- [7] W. W. Lee, J. Comput. Phys. 72, 243 (1987).

3D Gyrofluid Simulation of Electrostatic Ion Temperature Gradient Turbulence in a Sheared Slab*

W. Dorland, G. W. Hammett
Princeton University Plasma Physics Laboratory
P. O. Box 451, Princeton, NJ 08543 USA

July 1, 1991

In this work we investigate low-frequency, electrostatic turbulence driven by gradients in the ion temperature. So-called Ion Temperature Gradient (ITG) turbulence is suspected to contribute to the anomalous heat transport observed in tokamak experiments. ITG turbulence in a sheared slab is also a relatively simple example of a broader class of instabilities to which we hope to extend this research in the future. The linear stability and non-linear behavior of the ITG mode have been investigated with gyrokinetic and fluid models (among others). Near marginal stability, linear gyrokinetic models predict growth rates for the fastest growing mode roughly four times smaller than previous fluid simulations and average mode widths about two times smaller. Simple mixing length arguments therefore predicted a χ which was an order of magnitude less - a result confirmed by nonlinear gyrokinetic calculations¹. The discrepancies arose because these fluid models lacked a sufficiently accurate model for Landau damping and for finite Larmor radius effects. We present here a set of non-linear, electrostatic gyrofluid equations with an improved Landau damping² and FLR models suitable for 3D simulation of ITG turbulence in a sheared slab. The results of this approach and the gyrokinetic approach are compared.

Our equations are derived by taking moments of the gyrokinetic equation in the collisionless limit to yield evolution equations for n , v_{\parallel} , p_{\parallel} , T_{\perp} , and q_{\parallel} . The moment approach reduces the complexity of the problem by considering the evolution of only a few velocity space moments of the distribution function, instead of the full $F(v_{\parallel}, v_{\perp}, x, t)$. The moment hierarchy is closed in the parallel direction with a Landau damping model equivalent (in the present case) to a three pole approximation of the plasma dispersion function. Keeping more moments allows one to obtain systematically an n -pole approximation. This model effectively describes kinetic effects such as Landau damping and its inverse and is easy to implement in existing fluid codes to extend them to include wave particle effects. In the perpendicular direction, a similar closure equivalent to a two pole approximation accounts for the parallel phase mixing of perpendicular temperature. The electrostatic potential is calculated from the quasineutrality constraint resulting from Poisson's equation. The limitations of our approach and the related issue of how many moments need to be kept to achieve a desired level of accuracy are under investigation.

Assuming the usual gyrokinetic ordering, the non-linear gyrokinetic equation in the electrostatic limit is³

$$\frac{\partial F}{\partial t} + \nabla \cdot (F(v_{\parallel} \hat{b} + J_0 \nabla \times \hat{b})) - \frac{\partial}{\partial v_{\parallel}} \left(\frac{e}{m} F \hat{b} \cdot J_0 \nabla \phi \right) = 0.$$

*This research supported by DoE contract No. DE-AC02-76-CHO-3073

where J_0 is an operator which performs the gyroaveraging of the fields. The Fourier-space representation of J_0 is just the Bessel function with argument $k_{\perp} v_{\perp} / \Omega$. We define the moments in the usual way:

$$n \equiv \int F d^3v \quad n_{v_{\parallel}} \equiv \int F v_{\parallel} d^3v \quad p_{\parallel} \equiv m \int F (v_{\parallel} - v_{\parallel 0})^2 d^3v \\ T_{\perp} \equiv m/2 \int F v_{\perp}^2 d^3v \quad q_{\parallel, \perp} \equiv m/2 \int F (v_{\parallel} - v_{\parallel 0}) v_{\perp}^2 d^3v.$$

In the course of taking moments of the gyrokinetic equation we are faced with integrals such as:

$$\int d^3v F J_0 \nabla \times \hat{b} \equiv n(J_0) \nabla \times \hat{b}$$

In which we take the distribution function to be bi-Maxwellian with n , T_{\perp} , and T_{\parallel} functions of position. With these assumptions, we can perform these integrals, formally obtaining

$$(J_0) = \exp\left(\frac{1}{2} \rho_{\perp}^2 \nabla_{\perp}^2\right) \approx \exp\left(\frac{1}{2} \rho_{\perp}^2 \nabla_{\perp}^2\right) + \frac{T_{\perp}}{T_{\perp 0}} \frac{1}{2} \rho_{\perp}^2 \nabla_{\perp}^2 \exp\left(\frac{1}{2} \rho_{\perp}^2 \nabla_{\perp}^2\right) \nabla_{\perp}^2,$$

where the ∇_{\perp}^2 operates only on functions to the right. ρ_{\perp}^2 has been separated into its equilibrium and perturbed components using the fact that $T_{\perp 1} \ll T_{\perp 0}$ in the gyrokinetic ordering, while still allowing the strong nonlinearity $\nabla_{\perp} T_{\perp 1} \sim \nabla_{\perp} T_{\perp 0}$. Poisson's equation is treated analogously (a term appears in which the J_0 operates on the distribution function instead of a function ψ to the right). Our approximation for (J_0) is rigorously second-order accurate in $k_{\perp} \rho$ for any F (not just a bi-Maxwellian F) and is reasonably behaved even for large $k_{\perp} \rho$, unlike the Taylor-series approximation $(J_0) \approx 1 + \rho_{\perp}^2 \nabla_{\perp}^2 / 2$ inherent in most previous fluid equations which caused very large errors at large k_{\perp} and made the equations fairly stiff for numerical simulation.

The moments and the potential are then separated into equilibrium and fluctuating components. The resulting set of equations is cast in non-dimensional variables by normalizing the perpendicular directions to ρ_s , the parallel direction to L_n , the time to L_n / c_s , and the field variables as follows:

$$n = \frac{n}{n_0} \frac{L_n}{\rho_s} \quad v_{\parallel} = \frac{v_{\parallel}}{c_s} \frac{L_n}{\rho_s} \quad p_{\parallel} = \frac{p_{\parallel}}{p_{\parallel 0}} \frac{T_{\perp}}{T_{\perp 0}} \frac{L_n}{\rho_s} \\ T_{\perp} = \frac{T_{\perp}}{T_{\perp 0}} \frac{L_n}{\rho_s} \quad q_{\parallel, \perp} = \frac{q_{\parallel, \perp}}{c_p T_{\perp 0}} \frac{T_{\perp}}{T_{\perp 0}} \frac{L_n}{\rho_s} \quad \phi = \frac{e}{T_{\perp 0}} \frac{L_n}{\rho_s}$$

We set the present simulations in the standard sheared-slab geometry. Denoting the gyroaveraged potential $(J_0)\phi$ as Ψ , we use the following additional definitions:

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla \quad \mathbf{v} \cdot \nabla = (c/D) \hat{b} \times \nabla \cdot \Psi \\ r = \frac{r}{r_s} \quad J_0 = J_0(r \hat{k})$$

In comparing the equations below with previous fluid equations, it is important to note that our n is the guiding-center density, and not the actual particle density. There is a second gyro-center quantity in terms 5 and 6 of the Poisson equation which translates from guiding-center quantities into the particle density. The simulation is initialized with very low-level random fluctuations in the potential. We integrate the equations forward in time with a simple two-step explicit algorithm which is second-order accurate in time. The fields are represented spectrally in y and x (the directions of the sheared-slab magnetic field), and with a discrete grid in z .

Evaluation of nonlinear terms such as $\mathbf{v}_\phi \cdot \nabla n$ presently involves convolutions over the y and z Fourier-modes, although this needs to be upgraded to a pseudo-spectral algorithm to efficiently deal with larger systems. The gyro-averaging operator (J_0) is evaluated using FFT's.

Neglecting the parallel non-linearity, but including the lowest order non-linear FLR terms and the $\mathbf{E} \times \mathbf{B}$ non-linearities (the $\mathbf{v}_\phi \cdot \nabla$ terms included in the time derivative), we obtain:

$$\frac{dn}{dt} + \frac{1}{2}(\nabla_1^2 v_\phi) \cdot \nabla T_1 + \mathbf{b} \cdot \nabla u_{||} + \left(1 + \eta \frac{r}{2} \nabla_1^2\right) \frac{\partial \psi}{\partial y} = 0$$

$$\frac{du_{||}}{dt} + \frac{1}{2}(\nabla_1^2 v_\phi) \cdot \nabla q_{||,1} + \mathbf{b} \cdot \nabla (p_{||} + \psi) = 0$$

$$\frac{dp_{||}}{dt} + \frac{r}{2}(\nabla_1^2 v_\phi) \cdot \nabla T_1 + 3r\mathbf{b} \cdot \nabla u_{||} + r \left(1 + \eta \left(1 + \frac{r}{2} \nabla_1^2\right)\right) \frac{\partial \psi}{\partial y} + \underbrace{|\mathbf{k}_\perp|^2 \sqrt{\frac{2r}{\pi}} (p_{||} - nr)}_{^2} = 0$$

$$\frac{dT_1}{dt} + \frac{r}{2}(\nabla_1^2 v_\phi) \cdot \nabla (nr + T_1) + \mathbf{b} \cdot \nabla q_{||,1} + r \left(\frac{r}{2} \nabla_1^2 + \eta \left(1 + r \nabla_1^2 + \frac{r^2}{4} \nabla_1^4\right)\right) \frac{\partial \psi}{\partial y} = 0$$

$$\frac{dq_{||,1}}{dt} + \frac{r^2}{2}(\nabla_1^2 v_\phi) \cdot \nabla u_{||} + \mathbf{b} \cdot \nabla \left(\underbrace{\frac{2r}{\pi} T_1 + \frac{r}{\pi} (p_{||} - nr)}_{^3} + \frac{r^2}{2} \nabla_1^2 \psi\right) + \underbrace{|\mathbf{k}_\perp|^2 \sqrt{\frac{2r}{\pi}} q_{||,1}}_{^4} = 0$$

Poisson's equation (assuming quasineutrality and adiabatic electrons) is:

$$\left(1 + \frac{(1 - \Gamma_0(rk^2))}{r}\right) \psi = (J_0)_0 \underbrace{\left(n + \frac{r}{2} \nabla_1^2 T_1\right)}_{^5}$$

¹ Non-linear FLR terms.

² 3-pole model for parallel phase mixing of T_1 (which gives Landau damping).

³ Non-dissipative part of $\langle (u_{||} - u_d)^2 \rangle$ closure.

⁴ 2-pole model for parallel phase mixing of T_1 .

⁵ Polarization density.

⁶ Approximation to $\int d^3v J_0 F_1$ includes perturbations in T_1 .

Figure 1

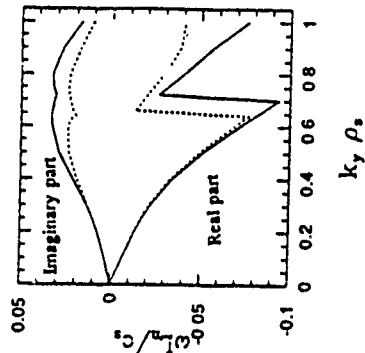


Figure 2

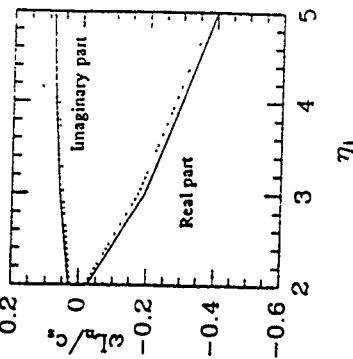


Figure 1 shows the linear behavior of the frequency of the fastest-growing even mode vs. $k_y \rho_s$ for our gyrofluid model and for an integral gyrokinetic code with $\eta = 2$, $T_e = T_i$, and $L_s = 40L_n$. The agreement is very good for small $k_y \rho_s$ and is acceptable even for large $k_y \rho_s$. Figure 2 shows the linear frequency for the fastest-growing even mode vs. η for $k_y \rho_s = 1/\sqrt{2}$, $T_e = T_i$, and $L_s = 40L_n$. Again, the results are very encouraging. Non-linear results will be presented for a variety of parameters to investigate the scaling of the thermal conductivity, χ .

In each figure, the solid line is the frequency obtained with the integral gyrokinetic code, and the dashed line is the result obtained with our gyrofluid code.

Acknowledgements: Thanks to Dr. S. H. Maguchi and Prof. W. Horton for providing the initial code from which our present code has evolved. Also, thanks to Dr. F. W. Perkins, Dr. R. E. Waltz, and Dr. W. Park for helpful discussions, and J. V. W. Reynders and Dr. W. W. Lee for the gyrokinetic comparisons. We would also like to acknowledge the very helpful *Multimath* and *Maple* computer programs which simplified the derivation of the local dispersion relation from the fluid equations.

- [1] M. Kotchenreuther et al., Thirteenth Int. Conf. on Plasma Physics and Controlled Nuclear Fusion Research (Washington, D.C., 1990).
- [2] G. W. Hammett and F. W. Perkins, Phys. Rev. Lett. **64** (1990) 3019.
- [3] E. A. Freiman and Liu Chen, Phys. Fluids, **25** (1982) 502. W. W. Lee, Phys. Fluids, **20** (1983) 556.
- [4] R. Linaker, Phys. Fluids **24** (1981) 1485.

Binary Collision Model in Gyrokinetic Particle Simulation

by MA, R.D. Sydora and J. M. Dawson

Department of Physics

University of California, Los Angeles

§1 Model

Binary collisions can be included in the gyrokinetic simulation model by using the Monte Carlo method^(1,2). The principle motivation for including binary collisions is to have a general collision scheme without regard to assumptions about background being Maxwellian, or if flows are present in plasmas. In such a combined model, the collective interaction and small-angle binary collisions are treated as two separate, uncoupled processes. First of all, the collision rate due to the use of an insufficient number of simulation particles is greatly reduced by assigning the particles a finite-size shape. This facilitates the study of collective properties of a plasma. The binary collisions are then included as follows.

1. The simulation system is divided into a number of spatial cells with a size such that plasma properties across each cell do not vary substantially. A typical size is the Larmor radius of the system in the case of gyrokinetic simulations. Particles are sorted into the cells by their actual positions, obtained by assigning each of them a random gyrophase.

2. Particles in each cell are paired randomly

3. Small angle collisions are performed pairwise. A collision between the two particles in a pair essentially involves a rotation of their relative velocity, while their positions are not affected by the collision. This rotation is characterized by a small scattering angle which obeys a Gaussian distribution whose mean squared deviation is determined by the product of the simulation time step and a collision frequency, and a polar angle which is uniformly distributed on $(0, 2\pi)$. The change of velocity for each particle is then obtained

4. The guiding center of each particle is shifted as a result of the velocity scattering

$$\Delta \mathbf{r}_g = \pm \frac{\Delta \mathbf{v} \times \mathbf{b}}{\omega_c}$$

where $\Delta \mathbf{v}$ is the change of velocity due to the collision, $\mathbf{b} = \frac{\mathbf{B}}{B}$ is the unit vector of the magnetic field, ω_c is the cyclotron frequency, the \pm sign accounts for the charge of the particle

It is clear that this binary collision model conserves the number of particles, the total energy and the total momentum pairwise and quasilocally. It can be shown that in the case of a uniform magnetic field, the guiding center shift scheme ensures an ambipolarity for cross-field particle diffusion due to unlike-particle collisions, while for like-particle collisions the center-of-mass of the particles is not changed

§2 Implementation

The original implementation of the binary collision model proposed in Ref. [1] is known to be inefficient. Here we present two schemes that take advantage of the present-day computer's capability of vector processing: a layered decomposition scheme and a linked-list scheme. Here vectorization is essential for the desired speedup. Since additional memory is often needed for vectorization, vectorization with minimum memory is necessary. It is easily seen that Step 3 of the collision model contains most of the floating-point calculations, which may be vectorized through indirect addressing. However the sorting of the particles into the cells is difficult, sometimes even inefficient to fully vectorize. In the layered-decomposition scheme, the particles are decomposed into a number of layers whose size is equal to the number of the cells used for sorting particles. Each layer contains up to one particle from each cell, and collisions are performed layerwise. This scheme can achieve a relatively high degree of vectorization, but its memory requirement can sometimes be intolerable in the case of inhomogeneous plasmas, where many layers are just sparsely filled. The linked-list scheme overcomes this drawback. The idea is to sort particles by using a pointer array which links particles in each cell, making full use of the additional memory for vectorization of Step 3. This scheme is not as vectorizable as the layered-decomposition one. Nevertheless, since Step 3 is vectorized anyway, the overall performance is greatly improved. This point is illustrated by the timing results shown below:

	Layered-Decomposition	Linked-List
Sorting	1.3 μ s	1.9 μ s
Pairing	0.5 μ s	1.5 μ s
Vel Rotation	1.5 μ s	1.6 μ s

where the CPU time per collision on a single processor Cray-2 is shown. In gyrokinetic simulations, the assignment of a random gyrophase and the guiding center shift requires

evaluation of the local magnetic field at the guiding center. On a Cray-2, this takes 2-4 μ s per pair of particles. The overall CPU per collision in gyrokinetic simulation is thus about 5-7 μ s with the layered decomposition scheme, and 7-9 μ s with the linked-list one.

§3 Tests and Applications

To test the velocity scattering part of the collision model, we have performed a number of tests including relaxation rates in velocity space, thermal equilibration and isotropization, and good agreement between simulation and test particle theory has been achieved. As an illustration, the result of the slowing-down of a test particle moving in a thermal medium is shown in Fig. 1, where the slowing-down rate is plotted vs. the energy of the test particle, shown by the squares. The solid curve is obtained from the test particle theory.

A simplest application of the gyrokinetic simulation model, including collisions, is the cross-field collisional transport of particles and heat in a uniform, fixed magnetic field. We have examined the classical transport due to ion-ion collisions. The result of the relaxation of an inhomogeneous plasma is shown in Fig. 2(a)-(b). It is seen that the temperature profile indeed relaxes much faster than the density, as is predicted by classical transport theory.

This model has also been employed to simulate neoclassical diffusion in a toroidal magnetic field configuration. The emphasis is laid on transport enhancement due to the toroidal-drift-induced charge separation, and on the contribution of like-particle collisions to particle and heat transport.

References

- [1] T. Takizuka and H. Abe, *J. Comp. Phys.* 25, 205 (1977)
- [2] S. Ma, R. Sydora and J. Dawson, *Bull. Am. Phys. Soc.* 35, 1982 (1990)

Acknowledgement

The authors would like to thank Dr. B. Cohen for useful discussions on implementation of binary collisions in gyrokinetic particle simulation model. This work was supported by DOE under Contract DE-FG03-86ER53223.

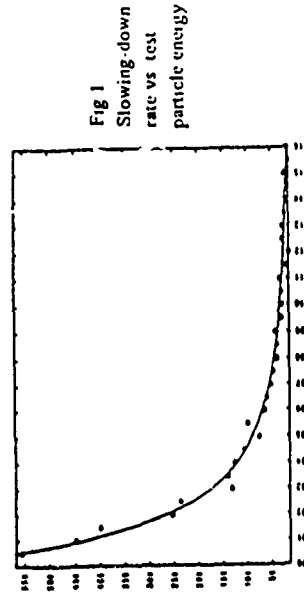


Fig 1
Slowing-down
rate vs test
particle energy

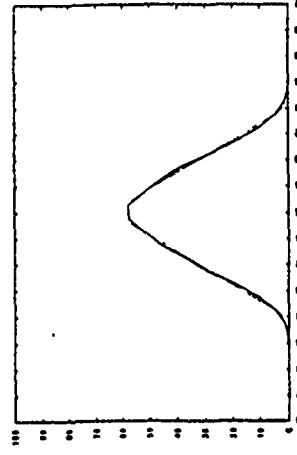


Fig 2(a)
Density vs x
Dots t=0
Curve t=10t_{ii}

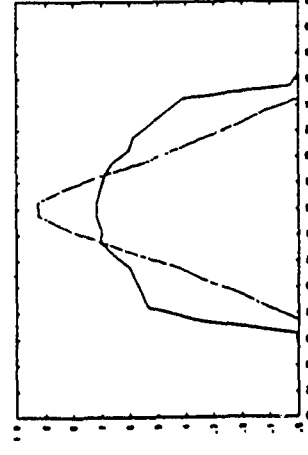


Fig 2(b)
Temperature vs x
Dots t=0
Curve t=10t_{ii}

3D Electrostatic Gyrokinetic Particle Simulation of Impurity Transport

by

Robert A. Santoro and W. W. Lee

Princeton University

The effects of plasma impurities on ion temperature gradient drift instabilities as well as the resulting particle and energy transport have been investigated using a 3D gyrokinetic particle simulation code in a sheared slab geometry [W. W. Lee, JCP 72, 243 (1987)]

To account for the various impurity concentrations, masses, and atomic numbers (Z), a new gyrokinetic Poisson equation has been derived. Impurities currently being studied are He^{2+} , C^{6+} , F^{9+} , and cold hydrogen ions. As expected, for small charge concentrations (< 1 %), the impurities seem to have no effect on the background turbulence while for higher concentrations (~ 5 %) there are noticeable effects. The Gyrokinetic Poisson Eqn in GK units (Ω^{-1} , ρ_s , $c\phi/T_e$) is given by:

$$(c_s + \frac{m_I}{m_i} c_I) \nabla_{\perp}^2 \phi = -Z n_I - n_i + \phi,$$

$$c_s = n_o/n_o, c_I = n_{oI}/n_o, c_s + Z c_I = 1, n_o = n_i + Z n_{oI}.$$

Along with the Poisson equation the gyrokinetic Vlasov equation must also be solved for both the bulk ions and the impurities. This equation may be written as:

$$\begin{aligned} \frac{\partial F}{\partial t} + v_{\parallel} \hat{B} \cdot \frac{\partial F}{\partial R} - \left(\frac{q}{m\omega} \right) \frac{\partial \Psi}{\partial R} \times \hat{B} \times \frac{\partial F}{\partial R} - \left(\frac{q}{m} \right) \frac{\partial \Psi}{\partial R} \cdot \hat{B} \frac{\partial F}{\partial v_{\parallel}} \\ + \left(\frac{q}{m\Omega} \right) \frac{\partial \Psi}{\partial R} \cdot K F = 0 \end{aligned}$$

where

$$\begin{aligned} \Psi(R) &\equiv \langle \phi \rangle (R) - \left(\frac{q}{2T} \right) \left(\frac{v_{\parallel}}{\Omega} \right)^2 \frac{\partial \phi(R)}{\partial R} \\ K &= \left[\kappa_{\parallel} - \left(\frac{3}{2} - \frac{v^2}{2v_{\parallel}^2} \right) \kappa_T \right] \end{aligned}$$

$F = F(R, \mu, v_{\parallel}, t)$ is the gyrophase-independent distribution function, $2\mu = v^2, \Omega = \frac{qB}{mc}$, B is the external magnetic field, $v_{\parallel} = (\hat{B} \cdot \nabla) \phi(R)$ is the electrostatic potential at the guiding center, and q is the charge. $\kappa_T = -d \ln T_e / dx$, $\kappa_{\parallel} = -d \ln n_i / dx$, and $E = -\nabla \phi$.

An important issue in Tokamak plasma transport is that of the anomalous inward pinch. The inward particle flux for the impurities has been observed to occur in the linear stage of the instability and to persist long after the saturation at a level somewhat lower than the quasilinear value. This flux consists of two components: a diffusive component and an inward pinch.

In order to estimate the inward pinch we consider moments of the gyrokinetic Vlasov equation for both the bulk ions and the impurity ions and so obtain the gyrofluid equations.

Gyrofluid Equations for the Bulk Ions:

$$\begin{aligned} \frac{dn_i}{dt} + \frac{\partial U_i}{\partial x_{\parallel}} + c_I \frac{\partial \phi}{\partial y} \left[\kappa_{\parallel i} + \frac{\nabla_{\perp}^2}{\tau} (\kappa_{\parallel i} + \kappa_T) \right] &= 0, \\ \frac{dU_i}{dt} + \frac{1}{\tau} \frac{\partial p_i}{\partial x_{\parallel}} + c_I \frac{\partial \phi}{\partial x_{\parallel}} &= 0, \\ \frac{dp_i}{dt} + c_I \frac{\partial \phi}{\partial y} [(\kappa_{\parallel i} + \kappa_T)] &= 0, \end{aligned}$$

Gyrofluid Equations for the Impurity Ions:

$$\begin{aligned} \frac{dn_I}{dt} + \frac{\partial U_I}{\partial x_{\parallel}} + c_I \frac{\partial \phi}{\partial y} \left[\kappa_{\parallel I} + \frac{\nabla_{\perp}^2}{\tau_I} (\kappa_{\parallel I} + \kappa_T) \right] &= 0, \\ \frac{dU_I}{dt} + \frac{m_I T_I}{m_i T_i} \frac{\partial p_I}{\partial x_{\parallel}} + Z \frac{m_i}{m_I} c_I \frac{\partial \phi}{\partial x_{\parallel}} &= 0, \\ \frac{dp_I}{dt} + c_I \frac{\partial \phi}{\partial y} [(\kappa_{\parallel I} + \kappa_T)] &= 0, \end{aligned}$$

where $\tau \equiv T_e / T_i$ and $\tau_I \equiv Z^2 (T_e / T_i) (m_i / m_I)$.

Using the gyrofluid equations for the bulk ions along with Poisson's equation an eigenmode equation can be found. The fundamental eigenmode is of the form:

$$\phi = \phi_0 e^{-i\omega t + i k y}$$

where

$$\sigma = \frac{\omega_{TI} L_{Ti}}{\omega L_e}$$

and ω is the eigenfrequency.

Next using the gyrofluid equations for the impurity ions to obtain the density response the average inward velocity of the impurities is found to be:

$$v_{inward} = \frac{\Gamma_I}{c_I} = -k_y \left(\frac{L_e}{L_{Ti}} \right)^{1/2} \left[Z_T \frac{m_I}{m_e} f \left(\frac{L_{Ti}}{L_e} \right) - \left(\frac{\omega_{TI}}{\omega T_I} \right) g \left(\frac{L_e}{L_{Ti}} \right) \right] |\phi|^2,$$

where $\omega_{TI} = k_y v_{TI}$, $\omega_{TI} = k_y v_{TI}$, and f and g are functions of the indicated arguments

The first term in the expression is the pinch and the second gives the diffusion. The pinch term comes from the parallel acceleration in the momentum equation for the impurities. The ambipolarity requirement for the particle flux yields $\Gamma_e = \Gamma_i + \Gamma_I = 0$ for this model, because the electrons are assumed adiabatic. Thus, the particle flux for the bulk ions are outward with a much smaller average velocity.

Simulation results have shown that the magnitude of inward pinch is larger than the diffusion for typical tokamak parameters. In addition, the inward velocities for the impurities studied were found to be quite similar even though the masses and charge states were significantly different. This is due to the dependence of the pinch velocity on $Z \frac{m_I}{m_e}$ which is similar for the cases studied.

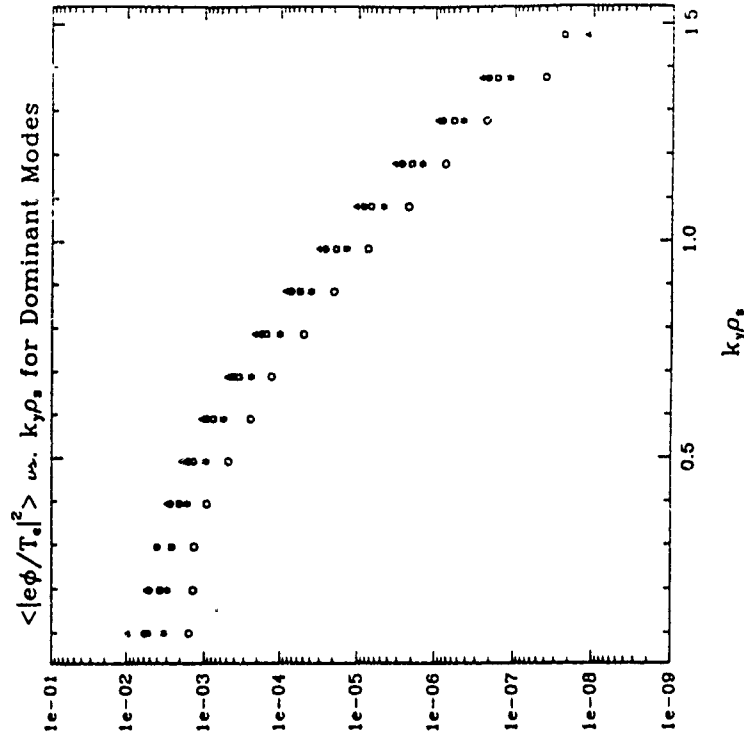
The nonlinear stage of the simulation has also been considered in detail. The magnitude of the inward particle flux of the impurities does not appear to be simply related to:

$$\sum |\phi_k|^2$$

but rather it may be related to the phase differences between the various modes ϕ_k in the system. Comparisons will be made between the flux level predicted by the particle code and quasilinear theory as well as other non-linear theory.

Finally, general particle and energy transport will be considered. The important question of system size scaling of anomalous transport in tokamaks will be addressed.

The Connection Machine II will be used test this scaling for very large systems. Current results indicate a weak scaling with system size.



Plasma formation by beam ionisation and collisions of beams focused by such a plasma*

S. Rajagopalan¹, C.-K. Ng² and P. Chen²

¹ Physics Department, University of California, Los Angeles, CA 90024

² Stanford Linear Accelerator Center, Stanford University, Stanford, CA 94309

Introduction

Plasma lenses for significant luminosity enhancement have been topics of keen interest in the recent past and currently.⁽¹⁾ They present a solution to the limits for spot sizes attainable by conventional magnet technology, and thus are strong candidates as final focus elements for linear colliders. The chief problem for high energy physics with plasma lenses is the background from the interaction of the high energy beam with the plasma, as well as that of the synchrotron radiation accompanying the beam. Thus the amount of material in the path of the beam has to be minimised. Further the production of plasmas at the densities of interest requires bulky devices like lasers or spark discharges. One option which provides a solution to working with the plasma requirements and the small volumes available at the interaction point in colliders is to produce the plasma by the ionisation of a gas target by the beam itself.

We first review the basic physics of the ionisation process and examine the specific example of the beam at the SLC (Stanford Linear Collider), the only high energy linear collider in the world at the moment. Then we proceed to describe the modifications required in a 2-d code CONDOR to implement the ionisation process and continually creating new particle species during the computation. We expect to present the results on the enhanced luminosities attained at the meeting.

Ionisation processes

The ionisation of atoms (targets) by energetic charged particles is a fundamental physical process which has been studied since the beginning of the century. The understanding of this process and the corresponding energy loss behaviour is the basis of most types of charged particle detection and analysis systems in high energy physics. The basic expressions for the cross sections for ionisation of a target by an energetic charged particle were worked out by H. Bethe. This cross-section for energetic particles has a dependence which is logarithmic with respect to energy and is of the order of a megabarn. This ionisation cross-section can

be written in the form⁽²⁾ where the target is considered to be a set of oscillators responding to the impulse of the electron. The expression is

$$\sigma = Ax_1 + Bx_2$$

where

$$x_1 = \beta^{-2} \ln(\beta^2/(1 - \beta^2)) - 1, x_2 = \beta^{-2}$$

and β is velocity of the electron in units of the velocity of light. The constants A and B are described by Inokuti⁽³⁾ as also the approximations involved in their determination. The cross-section for excitation of a target atom from the ground state to a state n is given by

$$\sigma_n = 4\pi(h/mc)^2(M_n^2 x_1 + C_n x_2),$$

M_n^2 and C_n are expressed in terms of the oscillator strengths of the transition. By integrating over the continuum we can write the ionization cross-section as

$$\sigma = 4\pi(h/mc)^2(M^4 x_1 + C x_2).$$

The table gives some sample values of M and C . The cross-section can be easily computed from the expression with these.

Values of M^2 and C for some gases

Gas	M^2	s.d.	C	s.d.
He	0.738	0.032	7.056	.040
Ne	2.02	0.05	18.17	0.06
Ar	4.22	0.15	37.93	0.19
Kr	6.09	0.16	52.38	0.22
Xe	8.04	0.15	72.35	0.40
CO ₂	5.75	0.10	55.92	0.40
Heptand	25.1	0.46	250.2	1.6

The other process occurs when the atom is in a strong electric field. There is the possibility of the electron tunneling through the potential barrier and thus the atom is

* Work supported by DOE contract DE-AC03-76SF00515 and DE-AS03-88ER40381

ionised. The probability (per unit time) for ionization is given by^[6]

$$W = 4 \frac{\alpha^4}{\lambda_D^3} \frac{ec}{E R_p} \exp\left(-\frac{2\alpha^2}{3} \frac{c}{E R_p}\right)$$

where α is the fine structure constant, λ_D is the Compton wavelength of the electron, E is the d.c. field, I is the ionization potential of the atom and R_p is the ionization potential of an infinitely massive Hydrogen atom or Rydberg ($\alpha^2/2\lambda_D$) as it is called. The above formula is for an atom in a uniform field. This expression is valid when the electric field are much smaller than the atomic field about 34 volts per angstrom. In fact the ionization becomes even classically possible much below this field gradient when the atomic levels rise above the potential barrier.^[6] If we have a dense electron beam then in the vicinity of the beam the fields are enormous and the above mechanism should contribute to the ionization of a gas target. The time of passage of the beam is of the order of a picosecond which is much higher than the periods for the atomic motions. Therefore we can regard the phenomenon as ionization due to an external d.c. field acting for the time the beam is in the vicinity of the atom of interest. In the case of an electron beam passing through a gas target the field is not constant in time. Hence the above expression will only give an approximation.

Plasma formation for beam-focusing and Simulation

In a test of plasma focusing with the plasma produced by ionization from the front of the beam, the important quantity is the fractional ionization due to the beam front. If the gas is not fully ionized by the beam front but continues to be ionized throughout the passage of the beam the result is a large variation in focusing strength from the head to the tail of the beam and hence the beam is not focused to one place. The fractional ionization from a beam of cross-sectional area A with N_b electrons is

$$\frac{N_b \sigma}{A}$$

For the SLAC beam with $N_b = 3 \times 10^{10}$, and measuring σ (cross-section) in units of 10^{-18} cm^2 and A in μ^2 the fractional ionization is $3\sigma/A$. Since we desire total ionization by the front of the beam, say 10 percent, the result is a catch-22! To achieve the goal we have to focus the beam down to the size which is the result of plasma focusing. However at the beam densities now possible at the interaction point of the SLC (Stanford Linear Collider) the tunneling ionisation becomes the dominant process. Thus the front of the beam, and truly the front as only 10-15 percent of the beam is involved, ionises and produces the plasma, followed by the rapid response of the plasma to the strong beam fields resulting in extremely strong focusing of the beam.

It is thus necessary to study through simulation the beam-gas-plasma interaction involved. We use a 2-D code CONDOR developed at Lawrence Livermore Laboratory. The code is modified to allow the continual production of a charged species through ionisation, which then takes part in the electromagnetic interaction. The modification requires subroutines to produce the new species during the main loop calculation which at present is not available. The main task is a great deal of book-keeping as the tunneling ionisation is a cumulative process over time and thus can't be dealt with by a simple Monte-Carlo simulation as the collisional ionisation can be. The results are of interest not only as basic physics of intense beams, but the luminosity of the collisions between two beams is of interest in determining the practical use of such a device.^[6]

We would like to acknowledge the help of S. Brandon, K. Eppley and K. Ko with questions on CONDOR.

REFERENCES

1. P. Chen, Particle Accelerators 17,121 (1987)
2. F. F. Rieke and W. Prepejchal, Phys.Rev. 6, 1507 (1972)
3. M. Inokuti, Rev Mod. Phys, 43,297 (1971).
4. L. D Landau & E. M. Lifshitz, Quantum Mechanics Pergamon press
5. P.Chen, SLAC-PUB-5502 (1991).
6. C.-K. Ng, S. Rajagopalan and P. Chen, these proceedings.

ON SOLVING THE SELF-CONSISTENT PROBLEM FOR BOUNDARY WEAKLY IONIZED GAS WITH ADMIXTURES AT NON-LINEAR PRESENTATION

S.W. Temko, K."Temko, S.K. Kuz'min

Moscow Geologicheskii Prospecting Institute, Moscow B - 485, USSR

Boundary weakly ionized gas with admixtures is ionized cloud. Admixtures are large particles weighted in gas and have been formed by adhesion and coagulation. Cloud has formed by action of inner, outer and surface forces, which are nearly connected with each others [1]. The result of joint action of these forces is cloud geometrical form and dimension, which are able to change. Only small their changes (of small perturbation type) are taken into account. Geometrical form and dimensions of the cloud are not given a priori. The self-consistent problem is solved by using the space clusters statistical thermodynamics, proposed before by the authors [2-5]. The solution is obtaining under the boundary condition: 1) we purpose, that surface potential is high enough and only very quick particles are able to leave the cloud, bringing only small perturbation at cloud state; 2) we take into account the interaction between single particles and cloud surface. Particles interaction with each others and with cloud surface is described by potential functions, their numerical parameters, cloud geometrical form and dimensions we use methods of experiment planning, likelihood maximum, directed random search and computer experiment. Cloud free energy is used as likelihood function. Parameters of potential functions and cloud dimensions are consisted the set of numerical parameters. We consider the particles to be "washed spots" and describe them by distributions. The state of the cloud with admixtures is described by vector-density of particles distribution. Term "distribution" is used in Sobolev-Schwartz sense. We purpose the cloud being in quasistatical equilibrium state. This do not limit generality of consideration, because quick processes have small momenta and they accurate cloud state change only after long enough time interval. From the conditions of quasistatical equilibrium for every time interval we obtain a system of equilibrium integral equation, which is to be solved jointly with additional normalization conditions. Potential functions are kernels of the integral equations. Equilibrium integral equation has been obtained for the first time by one of us [3]. The system of equilibrium integral equations is reduced

to equivalent system of Prandtl type integral equations, which is to be solved by expanding into series upon Chebyshev polynomials. Coefficients of linear algebraic equations are using reduction method. Coexistence of the last method is proved by effective Cantorovich-Krilov criterium [6]. Particles distribution vector-densities at every point of cloud volume differ from each other by sets of numerical parameters. The vector-densities are consisted the vector-densities set. Using random walk method we build on the vector-densities set the distribution function of random vector-density with random set of numerical parameters. Moments of random vector-density are cloud statistical thermodynamics distribution functions. Using sequential approximation method we solve non-linear equations system of single particle movement at the cloud. The moment to stop the calculations is established by results of comparison of particle coordinate two sequential approximations norm with relative mistake. We move partial coordinates and momenta along their movement trajectory according obtained laws of single particles movement. Having put particles coordinates and momenta moved according the laws of movement we obtain distribution functions of cloud non-linear statistical thermodynamics. We obtain expression for cloud free energy and its conditional minimum under conditions established from physical and physical-chemical considerations. Conditions system of cloud free energy conditional minimization includes "non-hard normalization", which has been proposed before and examined by S.K. Kuz'min [7,8]. "Non-hard normalization" includes stoichiometrical equations of chemical transformations, elastic and non-elastic interactions, energy, heat, movement quantity and particles number balance equations and diffusion equations and, also, several conditions, established by results of analysis physical and physical-chemical nature of the cloud. We obtain the most preferable potential functions, their numerical parameters and cloud dimensions and form corresponding to the maximum of likelihood ratio. Having put obtained numerical parameters at preferable potential functions we obtain the "true" laws of real interactions at the cloud. Using Ljapunov stability criterium we establish enough maximum estimate of cloud lifetime under small perturbation of its state. We obtain surface tension and coefficients of activeness for the cloud and, also, state equation.

for boundary weakly ionized real gas with admixtures. The obtained vector-density defines real profiles of density in the cloud. The obtained potential of macroscopical field is continuous at the whole space including intermedia surface and has all the necessary asymptotic properties. Comparison of observed and calculated data gives their good enough coexistence. Calculations are made using authors Pascal programs. The present method to solve self-consistent problem has been proposed for the first time by the authors.

Reference

1. Monte-Carlo methods in statistical physics / Under editing of Marchuk G.I. and Mikhailov P.A. Moscow: Mir. 1982 (in Russian).
2. Temko K.W., Temko S.W. On equilibrium potential. Doklady AN USSR. 1966. Vol. 166. N 3. pp. 551-554 (in Russian).
3. Temko S.W. On stable, in Ljapunov sense, equilibrium distribution of charged particles in ionized gas cloud. Prikladnaja mehanika i tekhnicheskaja fizika. 1969. N 1. pp. 20-26 (in Russian).
4. Temko S.W., Kuz'min S.K. Foundation of space clusters statistical theory. Dep. VINITI from March 28th 1986 N 2142-886. 42 p. (in Russian).
5. Temko S.W., Kuz'min S.K. System of Gauss equilibrium equations. Differencialnye uravnenia. 1990. Vol. 26. N 12. pp. 2110-2121 (in Russian).
6. Cantorovitch L.V., Akilov G.P. Functional analysis. Moscow: Nauka. 1972 (in Russian).
7. Kuz'min S.K. On minimization problem of multikomponent potential theory // Modern problems in exact sciences. Moscow: UDN. 1977. pp. 17-20. (in Russian).
8. Kuz'min S.K. On multikomponent potential theory. Dep. VINITI from June 14th 1976. N 2425-76. 42 p. (in Russian).

NUMERICAL METHODS FOR SIMULATING PROCESSING PLASMAS

V. Vahedi, M. Surendra,
G. DiPeso, J. Verboncoeur
University of California, Berkeley
Berkeley, CA 94720

RF glow discharges and other processing plasmas are used extensively in the microelectronics industry. Self-consistent fluid equations have been used recently to study the structural features of RF and DC glows^{1,2}. However, since these discharges are inherently complex and the particle distributions are non-Maxwellian, there has been a considerable effort in developing self-consistent kinetic models without making any assumptions on the distribution functions^{3,4}.

In order to use particle-in-cell simulation codes for modeling collisional plasmas and self-sustained discharges it is necessary to include interactions between charged and neutral particles. Monte Carlo methods have been used extensively in swarm simulations^{5,6}. In many Monte Carlo schemes, the time (or distance) between collisions for each particle is calculated from a random number. This allows for efficient algorithms, especially when the null collision method is used⁷. This technique is however, not compatible with PIC simulations, since all particle trajectories are integrated simultaneously in time. Thus, the collision probability for the i th charged particle is calculated, based on the distance $\Delta t_i = v_i \Delta t$ traveled in each time step Δt , to be

$$P_i = 1 - \exp(-\Delta t_i \sigma_f(E_i)n)$$

where σ_f is the total collision cross section, E_i is the kinetic energy of the particles and n is the neutral density. A collision takes place if a uniformly distributed random number on the interval $[0, 1]$ is less than P_i . The null collision method can be incorporated into the collision model by picking a constant collision frequency ν' such that,"

$$\nu' \geq n \nu \sigma_f$$

Thus, the computational cost of calculating P_i can be avoided. The fraction of the total number of particles (chosen at random) in the simulation that experience collisions is given by

$$P = 1 - \exp(-\nu' \Delta t)$$

The collision is assumed to take place at the current position of the particle. It should be noted that the choice of Δt will affect the accuracy of the collision model. For instance, Δt_i should not be much larger than simulation length scales of interest ($e.g.$ grid spacing, λ_{De}) and $\Delta t_i \sigma_f(E_i)n$ should be about 0.1 or less¹⁷. Once a collision occurs, the type of collision, the energy of the ejected electron (for an ionizing collision) and the direction(s) are determined with new random

numbers. These quantities are related back to the system coordinate axes. The procedure for electron-neutral collisions is describe in detail by Boeuf and Marode¹⁸, and by Thompson *et al.*¹⁴ for ion-neutral collisions. Expressions for differential cross sections that are analytically integrable are useful as the computational cost of determining scattering angles and energy redistribution in ionizing collisions is minimized^{4,19,15,17}.

A Monte Carlo collision handler as described above, including the null collision method, has been developed as an addition to the PIC scheme as shown in Fig. 1. The full three dimensional character of a collision is modeled with three velocity components. The neutrals are assumed to be uniformly distributed between the boundaries with a constant density and a Maxwellian profile. The model is still valid if the neutral density is a weak function of position and time (small variations across the mean free path and collision times). This scheme can also be extended to model Coulomb collisions between charged particles.

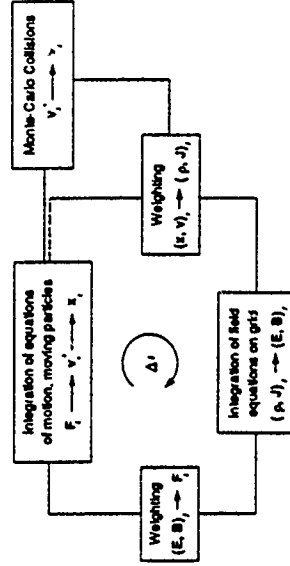


FIG. 1. The flow chart for an explicit PIC scheme with the addition of the collision handler, called PIC-MCC.

RF discharge modeling displays many physical time scales, e.g. $\omega_{pe} \ll \omega_{pi}$. With a PIC model including an electrostatic response, the highest frequency that must be resolved by the explicit numerical methods used to solve the particle and field equations is ω_{pi} . If $\omega_{pi} \Delta t > 1$, numerical instabilities can occur for explicit methods¹⁹. To observe the physics of interest, one needs to resolve the RF timescale only, and therefore much computing time is wasted resolving the plasma oscillation time scale.

Implicit particle simulation¹⁹ has been developed to relax the numerical stability constraint. We will now briefly review implicit particle simulation techniques. Implicit particle movers advance the position of the i th super particle by the equation

$$x_i^{n+1} = \alpha' E^{n+1}(x_i^{n+1}) + x_i^n$$

where \mathbf{x} is the portion of the position advanced, dependent on quantities known at present and previous time levels, $\alpha' = \beta \Delta t^2 q/m$, and β depends on the particular implicit scheme. The field at a particle location is interpolated from the field known at the grid in space. Note that the particle location at the future $n+1$ time step depends on the field at that time step, but the field at the future time step depends on the particle location at that time step through the Poisson equation.

One way to get around this problem is to linearize the locations \mathbf{x}_i^{n+1} about the locations \mathbf{x}_i^{n+1} ; in the superparticle-to-grid weighting equations. Then

$$\mathbf{p}^{n+1} = \tilde{\mathbf{p}}^{n+1} - \partial_i \tilde{\mathbf{p}}^{n+1} \delta \mathbf{x}_i^{n+1}$$

where $\tilde{\mathbf{p}}^{n+1}$ is determined by weighting superparticles at \mathbf{x}_i^{n+1} to the grid. The minus sign is from the functional dependence of the particle positions in the superparticle-to-grid weighting equations. Strictly speaking, $\delta \mathbf{x}_i^{n+1} = \mathbf{x}_i^{n+1} - \mathbf{x}_i^{n+1}$ is an individual quantity for each superparticle and is given by the advancing equation. Instead, $\delta \mathbf{x}_i^{n+1} = \delta \mathbf{x}_i^{n+1} - \alpha' E^{n+1}$, i.e., the perturbations are taken to be grid quantities while maintaining the form as given by the advancing equation. Inserting the above form of the perturbation into the equations for the density and combining with the Poisson equation gives the numerically implicit Poisson equation

$$\partial_i [1 + \alpha \tilde{\mathbf{p}}^{n+1}] \partial_i \tilde{\mathbf{p}}^{n+1} = -\tilde{\rho}^{n+1}/\epsilon_0$$

where $\alpha = \alpha'/\epsilon_0$.

The equation is solved on a spatial grid. Simple boundary conditions for the RF discharge are a zero potential at the left wall and an RF source voltage at the right wall. The electric field is found at the interior points by central differencing the potential. The electric field at the walls is given by a numerically implicit Gauss' law which is the integral form of the numerically implicit Poisson equation. The electric field, at the left wall for example, is then derived by taking a Gaussian pill box about the wall,

$$[(1 - \alpha \tilde{\mathbf{p}})E]_{x=0} = \sigma_0/\epsilon_0 + \tilde{\rho}_0 \Delta x/2\epsilon_0$$

The enclosed charge (RHS) includes the wall surface charge density $\sigma_0 = \epsilon_0 E_0$ where the 0 subscript indicates the left wall quantity and $j=1/2$ indicates an evaluation between the 0 and 1 grid points. It is then possible to solve for the electric field on the left wall. A similar procedure is used for the right wall. The equations may be generalized to included more complicated boundary conditions including external circuit elements

Many accuracy constraints still remain. One important constraint is that the fastest particle species should resolve spatial gradients in the field, i.e., $v_{\text{max}} \Delta t/s < 1$ where s is the gradient length. Another important constraint is that all particles should sample the field on the grid in a continuous manner over a single time step. This gives $v_{\text{max}} \Delta t/\Delta x < 1$. A problem with implicit methods is excessive numerical cooling which is due to poor sampling of fast particles in simulations with large time steps. Resolving fast particles is particularly important in RF discharges because it is

the fast electrons which maintain the discharge through ionization collisions with the neutrals. A possible way out of this problem is to do multi-scale simulations. That is, the few fast electrons that maintain the discharge may be pushed with a small time step while the remaining slow particles, essentially residing in the bulk plasma, may be pushed with a large implicit time step.

ACKNOWLEDGEMENT

This work is supported in part by DOE Contract DE-FG03-90ER54079 and ONR Contract N00014-85-K-0809, and the San Diego Supercomputer Center.

REFERENCES

- ¹ D. B. Graves and K. F. Jensen, *IEEE Trans Plasma Sci* PS-14, 78 (1986)
- ² J. P. Boeuf, *Phys Rev A* 36, 2782 (1987).
- ³ E. Gogolides, J. P. Nicolai, and H. H. Sawin, *J Vac Sci Technol A* 7, 1001 (1989)
- ⁴ Birdsall C. K., *IEEE Trans Plasma Sci* 19, 65 (1991).
- ⁵ Surendra M., and D. B. Graves, *IEEE Trans Plasma Sci* 19, 144 (1991a)
- ⁶ V. Vahedi, M. A. Lieberman, M. V. Alves, J. P. Verboncoeur, and C. K. Birdsall, *J Appl Phys* 69, 2008 (1991).
- ⁷ Vender D. and R. W. Boswell, *IEEE Trans Plasma Sci* 18, 725 (1990).
- ⁸ Sommerer T. J., W. N. G. Hieken, R. E. P. Harvey, and J. E. Lawler, *Phys Rev A* 43, 4452 (1991)
- ⁹ S. R. Hunter, *Aust J Phys* 30, 83 (1977).
- ¹⁰ J. P. Boeuf and E. Marode, *J Phys D* 15, 2169 (1982)
- ¹¹ Y. Kaufman, *J Phys D* 21, 442 (1988).
- ¹² M. J. Kushner, *J Appl Phys* 58, 4024 (1985)
- ¹³ M. J. Kushner, *J Appl Phys* 54, 4958 (1983).
- ¹⁴ B. E. Thompson, H. H. Sawin and D. A. Fisher, *J Appl Phys* 63, 2241 (1988)
- ¹⁵ S. L. Lin and J. N. Bardsley, *J Chem Phys* 66, 435 (1977).
- ¹⁶ M. Surendra, D. B. Graves, and I. J. Morey, *Appl Phys Lett*, 56, 1022 (1990)
- ¹⁷ M. Surendra, D. B. Graves, and G. M. Jellum, *Phys Rev A* 41, 1112 (1990)
- ¹⁸ C. K. Birdsall and A. B. Langdon, *Plasma Physics via Computer Simulation* (McGraw-Hill, New York, 1983)
- ¹⁹ A. B. Langdon, B. I. Cohen, A. Friedman, *J Comp Phys* 51, 107 (1983)
- ²⁰ "Multi-Scale Particle Simulation of Bounded Plasmas", S. E. Parker, A. Friedman, S. L. Ray, and C. K. Birdsall, *Proc 13th Conf on Numerical Simulation of Plasmas*, Santa Fe, New Mexico (1989)

SOLUTION OF STRONGLY NONLINEAR DIFFUSION EQUATIONS¹

Glenn Bateman

Princeton Plasma Physics Laboratory, P. O. Box 451, Princeton, NJ 08543 USA

14th International Conference on the

Numerical Simulation of Plasmas

Annapolis, Maryland

3-6 September 1991

Two simple methods have been found to reduce the numerical instability associated with computing the time evolution of one dimensional heat diffusion equations in which the thermal diffusivity has a sharp threshold as a function of the temperature gradient.^[1] These methods involve

1. increasing the implicitness parameter θ to values well above unity (typically to 4.0 or more) in order to reduce overshooting; and
2. limiting the local rate of change of the diffusivity to avoid the rapid growth of spikes

It is found that these methods can be used together to provide a simple practical method for keeping the numerical instability under control, even when abrupt physical perturbations (such as sawtooth oscillation crashes in simulations of tokamak plasmas) periodically drive radical changes in the temperature gradient.

The numerical instability at issue here appears when the temperature gradient just equals the threshold value over an extended spatial region. Under these conditions, any perturbation that increases the temperature gradient at a point within this region causes the diffusivity there to increase to a high value (above the threshold). Then, on the next time step, the temperature gradient at that point is flattened and the diffusivity drops to a low value (below the threshold), while the temperature gradient on each adjacent gridpoint is increased sharply above the threshold. Hence, the temperature gradient and the diffusivity overshoots and undershoots on adjacent gridpoints. An example of this instability is shown in Figure 1. The same simulation using a combination of the methods described in this paper is shown in Figure 2. These figures are taken from a series of plasma transport simulations using a combination of theoretically-based thermal transport models to predict temperature profiles in the TFTR tokamak.^[1] The transport model that dominates in the core of the plasma turns on abruptly above a threshold ion temperature gradient and rises to large values above this threshold.^[2]

This numerical instability could be eliminated by taking sufficiently small timesteps. For the simulations shown here, the timesteps would have to be at least a hundred

¹Work supported by the U. S. DoE Contract No. DE-AC02-78-CH0-3073.

times smaller than otherwise needed, which would be impractical. Alternatively, it was found that spatially smoothing the profiles or their gradients just increased the wavelength of the instability without appreciably reducing its amplitude. The method described in this paper is not perfect — it decreases the accuracy of the simulated time evolution and it allows physically driven spikes in the temperature gradient to persist longer than they should. However, the method has allowed simulations that would otherwise not be feasible.

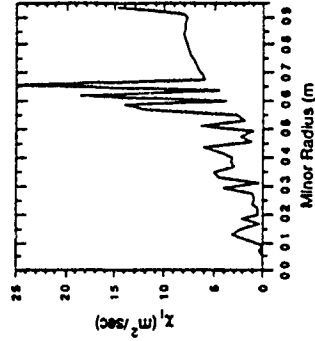


Fig. 1. Ion thermal diffusivity from a numerically unstable simulation.

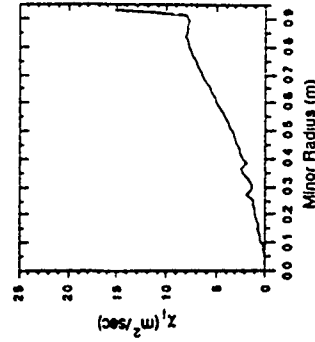


Fig. 2. Ion thermal diffusivity from a numerically stable simulation.

1. Implicitness Parameter Greater Than Unity

When advancing diffusion equations in time using implicit finite difference methods numerical analysis textbooks^[3] normally recommend the choice of implicitness parameter $\theta = 0.5$, (called the Crank-Nicholson method) because the linear diffusion equation is both numerically stable and second order accurate in time with this choice. For nonlinear diffusion equations (in which the diffusivity is a function of the solution of the equation), a greater margin of stability is usually needed and, consequently, values of θ between 0.6 and 1.0 are usually chosen. However, we were finding symptoms of numerical instability, as shown in Figure 1, with any choice of $0.5 < \theta \leq 1$, whenever the thermal diffusivity had a form which increased sharply as a function of the temperature gradient above a threshold value, as shown schematically in Figure 3.

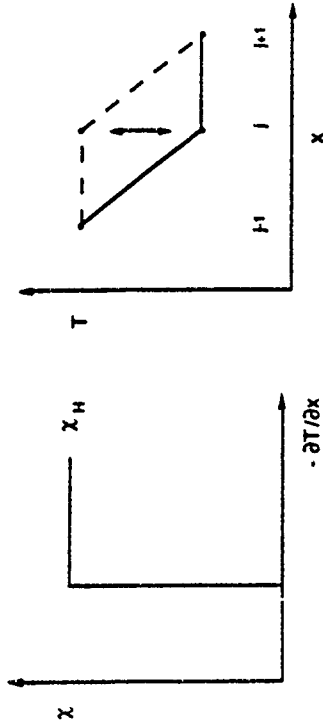


Fig. 3. Schematic of thermal diffusivity as a function of temperature gradient.

Consider a simple temperature diffusion equation

$$\frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \chi \frac{\partial T}{\partial x},$$

where the diffusivity χ has a sharp turn-on as a function of $(\partial T / \partial x)$, as shown in Figure 3. Consider three grid points (with the end points fixed) on which the temperature gradient goes from above the threshold to below the threshold, as shown in the lower (solid) curve in Figure 4. Then, an implicit finite difference scheme for the central grid point has the form

$$\frac{T_j^{N+1} - T_j^N}{\Delta t} = -\frac{\chi_H}{(\Delta x)^2} [\theta(T_{j-1}^{N+1} - T_{j-1}^N) + (1 - \theta)(T_j^N - T_{j-1}^N)]$$

For large values of $\chi_H \Delta t / (\Delta x)^2$ and fixed endpoints ($T_{j-1}^{N+1} = T_{j-1}^N$) we have

$$T_j^{N+1} \rightarrow \frac{T_j^N + (\theta - 1)T_{j-1}^N}{\theta}$$

For the choice of $\theta = 1$ (fully implicit), it can be seen that the value of T on the central grid point just oscillates up and down from one time step to the next, as shown in Figure 4. However, for the choice $\theta = 2$, after one time step the central grid point becomes the average of the two end points and the gradient becomes smooth. For values of $\theta > 2$, the central grid point rises more slowly and the steep part of

the gradient (on the left) approaches the threshold value more gradually. Hence, in a driven system with a steep threshold, taking the implicitness parameter $\theta \geq 2$ will allow the gradients to relax to the threshold value, damping out any overshoots.

In practice, this method does not work perfectly. Sharply defined processes such as sawtooth crashes or the noise from Monte Carlo computations tend to drive the temperature gradient well above the threshold in some locations transiently. Also, with long time-steps, the temperature gradient can increase well above the threshold during one time step. It then takes a while to relax back to a smoother profile.

2. Limit the Local Rate of Change of Diffusivity

In order to implement this method, a copy of the diffusivity array is kept from the previous timestep (χ^N), and the diffusivity is computed for the new timestep (χ^{N+1}). The idea is to limit the growth of local spikes while leaving the diffusivity free to change on larger scales. The following transform accomplishes this objective

$$\chi^{N+1} = \chi^N + \frac{\chi^{N+1} - \chi^N}{1 + C[\chi^{N+1} - \chi^N - (\chi^{N+1} - \chi^N)]^2},$$

where (...) represents spatial smoothing. (Here I use $A = (A_{j+1} + 2A_j + A_{j-1})/3$). χ^{N+1} is used in the diffusion equation and a copy is kept for the next time step. Choosing the constant $C = 10$, for example, limits the growth of diffusivity spikes to about 0.1 during each time step. The temperature gradient then has a chance to adjust gradually and damp out any tendency to overshoot.

A combination of these methods has allowed me to carry out simulations with extremely steep diffusivity thresholds. A problem with the second method is that the diffusivity used in the diffusion equation takes a while to catch up to the diffusivity computed from the physical processes whenever there are abrupt changes. A second problem is that the constant C has dimensions of one over diffusivity and must be chosen to suit the scale of the diffusivity being used. Hence, these methods are best suited to diffusion problems where the solution is needed where the conditions are changing gradually or approaching steady state.

- [1] Glenn Bateman, "Theory-Based Transport Simulations of TFTR L-mode Temperature Profiles," Princeton Plasma Physics Laboratory report PPPL-2764 (1991).
- [2] S. Hamaguchi and W. Horton, "Fluctuation Spectrum and transport from ion temperature gradient driven modes in sheared magnetic fields," Phys. Fluids B 2 (1990) 1833-1851.
- [3] R. D. Richtmyer and K. W. Morton, "Difference Methods for Initial-Value Problems," Interscience Publishers, John Wiley and Sons, NY (1967).

An Adaptive Grid with Directional Control for Toroidal Plasma Simulation

J. U. Brackbill
Los Alamos National Laboratory
Los Alamos, NM 87545

Introduction

One direction for the future development of plasma simulation is toward methods in three-dimensional, toroidal geometry like that shown in Fig. 1. The passing and trapped particle orbits are calculated in a torus with small aspect ratio and circular cross section. More realistic simulations require many more zones and particle orbits, and the computational power of highly concurrent processors.

The computation mesh can do much to increase the accuracy and decrease the cost of plasma simulation in toroidal geometry. The grid can conform to the boundaries of the physical domain to eliminate inactive regions[1]. The grid can adapt to resolve local gradients, or it can move to equalize the number of particles from zone to zone[2]. Here, we describe adaptive grids in three dimensions, and discuss recent work which allows one to orient a grid along magnetic field lines using a variational formulation and a structured grid[3].

There are advantages to aligning the grid and the magnetic field. Particle motion is predominantly along magnetic field lines, and points along field lines tend to have equal temperatures and pressures. If these fast time scale phenomena can be associated with one-dimensional arrays of cells, one can consider techniques that reduce the dimensionality of simulation in a torus from three to two.

Structured Grid

In a structured grid, the vertices of the grid x_{ijk} map to a logical grid (i, j, k) , and intermediate points, $ix_{\frac{1}{2}j+1, k}$, $jx_{i+1, \frac{1}{2}k}$, and $kx_{i, j+1, \frac{1}{2}}$ map to intermediate points, for example by trilinear interpolation,

$$x(\xi, \eta, v) = (1-v) \left[(1-\xi) \left\{ (1-\eta)x_{ijk} + \eta x_{i+1, jk} \right\} + \xi \left\{ (1-\eta)x_{ijk} + \eta x_{i+1, jk} \right\} \right] \quad (1)$$

which maps each six-sided cell in physical space on to a unit cube in logical space.

An Adaptive Body-Fitted Grid Generator

A mapping giving a body-fitted grid is generated by minimizing the functional[4],

$$I = \sum_{i=1}^N \iiint dx dy dz w |\nabla \xi|^2, \quad (\xi^1, \xi^2, \xi^3) = (\xi, \eta, v) \quad (2)$$

for which the Euler equations are,

$$\nabla \cdot \frac{1}{w} \nabla \xi^j = \frac{1}{w} \nabla \cdot \nabla \xi^j - \frac{1}{2} \nabla w \cdot \nabla \xi^j = 0, \quad j=1,2,3 \quad (3)$$

Eq. (3) is usually solved with the mesh points specified along the boundaries of the mesh.

A geometric interpretation of Eq. (3) suggests the properties of the solution. The right hand side of Eq. (3) is the projection of the gradient of w in a direction perpendicular to surfaces of constant ξ, η , and v , for $j=1,2,3$ respectively. Thus, the variation in spacing between level surfaces of the natural coordinates is proportional to the component of the gradient of w normal to the surface. When w is a constant, the solution is the smoothest mapping satisfying the boundary conditions.

A computationally convenient form of the generator equations is derived by interchanging dependent and independent coordinates so that differentiation is with respect to the natural coordinates. We introduce covariant and contravariant mesh vectors [1,5]. The covariant and contravariant mesh vectors are defined by,

$$a_i = \frac{\partial x}{\partial \xi^i}, \quad a^i = \nabla \xi^i = \frac{\partial}{\partial x} \times a_i \quad (4)$$

where $(i, j, k) = (1, 2, 3)$ in cyclic permutation and the Jacobian, J , is defined by,

$$J = (\det(g_{ij}))^{1/2} = a_1 \cdot (a_2 \times a_3), \quad (5)$$

where the covariant and contravariant metric tensors are defined by,

$$g_{ij} = a_i \cdot a_j, \quad g^{ij} = a^i \cdot a^j \quad (6)$$

The covariant vector a_1 , for example, is tangent to intersections of surfaces of constant η and v . The contravariant vector, a^1 , is perpendicular to ξ - constant surfaces. Using the orthogonality relation,

$$a_i \cdot a^j = \delta_i^j$$

and the identity,

$$\nabla \cdot \nabla x = 0,$$

one derives the generator equation,

$$\sum_{i=1}^3 g^{ij} \frac{\partial^2 x}{\partial \xi^i \partial \xi^j} = - \frac{\nabla w}{w} \quad (7)$$

The derivatives of the physical coordinates are evaluated by differentiating a mapping like that given in Eq. (1).

The Role of the Weight, w

The weight function, w , is defined by the user. If one wishes to vary the mesh spacing to resolve gradients of a variable U , one defines $w[2]$,

$$w = \frac{VU}{U}. \quad (8)$$

if one wishes to cause there to be a constant number of simulation particles, n , in each cell, one defines $w=n[6]$. For toroidal geometry, the generator equations must be modified to reflect the curvilinear coordinates of the torus. Alternatively, one may alter the weight function so that the Cartesian equations generate a toroidal mesh. For example, if one wishes to generate a polar mesh with r/i -constant, and θ/j -const., one defines $w=i$. Similarly, for toroidal geometry, one defines $w=iR$, where R is the major radius.

Directional Control

Giannakopoulos and Ennis have shown in two dimensions that a functional, I_d , where I_d is minimized when grid lines are aligned with a prescribed vector field, can be added to I_S to give directional control[3]. This approach can be generalized to three dimensions to cause grid lines to align with a magnetic field.

Consider a magnetic field for which one can define flux surfaces, ψ -const., in which lie magnetic field lines, $\mathbf{B} \cdot \nabla \psi = 0$. (This simplifies the discussion, but it is not necessary to the method.) We wish to generate a mesh in which ψ is constant on surfaces of constant v , and grid lines with tangent vector $\mathbf{x}\eta$ are aligned with the magnetic field. From Eqs. (4) and (5), one finds,

$$\mathbf{x}_\eta = [\nabla v \times \nabla \xi] \mathbf{j} \quad (9)$$

Since $\mathbf{B} \cdot \nabla \psi = 0$, minimizing the functional,

$$I_d = \int d^3\mathbf{x} [(\nabla \psi \times \nabla v)^2 + (\mathbf{B} \times \nabla \psi \times \nabla \xi)^2] \mathbf{j} \quad (10)$$

yields

$$\mathbf{x}_\eta = \alpha [\nabla \psi \times (\mathbf{B} \times \nabla \psi)] = \alpha \nabla \psi^2 \mathbf{B}. \quad (11)$$

the desired result. The functional can be written more conveniently,

$$I_d = \iiint d\xi d\eta dv \left[\left\langle (\rho \cdot \mathbf{x}_\eta)_\xi^2 - (\rho \cdot \mathbf{x}_\eta)_\eta^2 \right\rangle + \left\langle (\Theta \cdot \mathbf{x}_\eta)_\xi^2 - (\Theta \cdot \mathbf{x}_\eta)_\eta^2 \right\rangle \right] \quad (12)$$

where $\Theta = \mathbf{B} \times \nabla \psi$, and $\rho = \nabla \psi$. Minimizing the functional, $I = I_S + \lambda I_d$, with λ positive, results in a mesh with spacing determined by w and the vectors \mathbf{B} and $\nabla \psi$.

The composite Euler equations minimizing $I = I_S + \lambda I_d$ form a system of equations that couples the three coordinate directions. The matrix of coefficients is non-symmetric. Presently, the equations are solved using a conjugate gradient iteration on the equations in two dimensions, and a Jacobi iteration in three.

References

1. Thompson, J.F., Warsi, Z.U.A. & Mastin, C.W. *Numerical Grid Generation: Foundations and Applications* (Elsevier, New York, 1985).
2. Brackbill, J.U. & Saltzman, J.S. *J. Comput. Phys.* **46**, 342-367 (1982).
3. Giannakopoulos, A.E. & Engel, A.J. *J. Comput. Phys.* **74**, 422-439 (1988).
4. Brackbill, J.U. & Ruppel, H.M. *J. Comput. Phys.* **65**, 314 (1986).
5. D'haeseleer, W.D., Hitchon, W.N.G., Callen, J.D. & Shohet, J.L. *Flux Coordinates and Magnetic Field Structure* (Springer-Verlag, New York, 1991).
6. Brackbill, J.U. in *Computer Applications in Plasma Science* (eds. Drobot, A.) 422-457 (Springer-Verlag, New York, 1991).

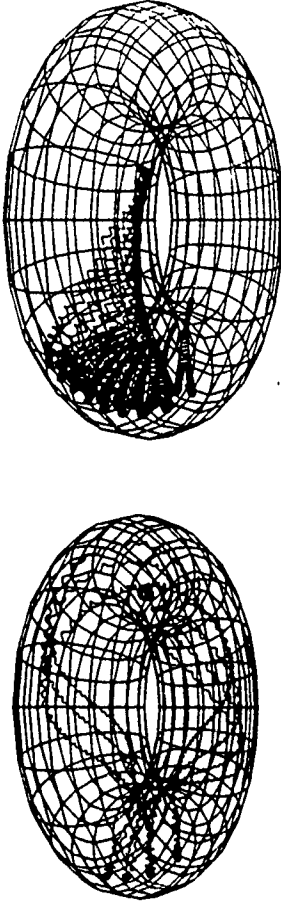


Fig. 1 Particle orbits in a toroidal magnetic field are superimposed on a plot of the outer surface of the computation grid on which the orbits are calculated. On the left, a passing particle orbit is plotted, and on the right a trapped particle orbit is plotted.

Three-Dimensional Eigenmode Analysis for Structures with Resistive Dielectrics and Arbitrary Boundary Conditions.

W.A. Krueger and A.T. Drobot
Science Applications International Corporation
McLean, VA 22102

We describe a new algorithm for finding the eigenvalues and eigenvectors, in the frequency-domain, of three dimensional structures. The method is useful for finding the complex resonances of enclosed cavities such as are commonly found in accelerators and microwave devices. The new method can also be applied to efficiently determine the dispersion relations for periodic structures.

To accurately resolve complicated three-dimensional structures of interest, the number of node points required can easily be on the order of 10^6 . The direct solution of the associated eigenvalue problem is prohibitive because of the size of the operator involved. The computational feasibility of these problems resulted from the introduction of a deflation scheme which was developed by Tuckmantel[1]. The scheme reduces the problem to finding the lowest few eigenvalues directly from a small submatrix. This method has been used by Wieland in MAFIA[2] and also in the ARGUS code[3] to compute real eigenvalues and eigenvectors.

Using the basic eigenvalue extraction scheme of Tuckmantel we have made several improvements that are incorporated in the new algorithm. We have simplified the extraction procedure to require a minimum input from the user and have automated the criteria to identify well converged solutions. We have extended the scheme to include complex dielectrics and resistive materials. This necessitated the extension of the basic algorithm to deal with complex eigenvalues. We have constructed the code so that it permits an arbitrary phase shift across a single cell of a periodic structure. This permits the determination of the dispersion relation for devices using just a single, well-resolved period of the structure. Finally, the stand alone version of the algorithm has been implemented on a Macintosh IIx, and takes advantage of the readily available visualization tools for this platform (e.g., NCSA Image).

Physical Model

The equations of interest in the frequency domain are the Maxwell equations written as,

$$\lambda \mathbf{B} = -\nabla \times \mathbf{E}, \quad (1)$$

$$\lambda \mathbf{D} = \nabla \times \mathbf{H} + \sigma \mathbf{E}, \quad (2)$$

where $\mathbf{B} = \mu \mathbf{H}$, $\mathbf{D} = \epsilon \mathbf{E}$ and λ is the eigenvalue we seek. The presence of the second term on the right-hand-side of the Eq. 2 leads to dissipation. Physically this term represents a resistive element in the cavity or the loss tangent of the dielectric which would lead to the damping of the mode and eigenvalues which have a negative real part.

The simple inclusion of a resistive term in the model subtly changes the property of the system, and this change necessitates a complete revision of the numerical procedure to be used.

Numerical Algorithm

The numerical algorithm is based on techniques developed by Tuckmantel. The underlying premise of the method is that a sequential application of the shifted Maxwell operator to a trial solution will project out a subspace which contains only a few lowest eigenvectors. An equivalent matrix problem on this subspace will have the same eigenvalues as the overall problem. In the presence of complex materials the eigenvalues of the Maxwell operator occur in complex pairs (λ, λ^*) or are purely real and negative ($-\lambda_1, -\lambda_2$).

To reduce a trial solution to a subspace requires a proper set of shift factors which come in complex pairs. We have used shifts of the form $s_n = \alpha_n + i\beta_n$ where α_n and β_n are sequential iteration parameters and are used as follows:

$$\begin{aligned} \mathbf{B}^{n+1} &= -\nabla \times \mathbf{E}^n + s_n \mathbf{B}^n, \\ \mathbf{D}^{n+1} &= \nabla \times \mathbf{H}^n - \sigma \mathbf{E}_n + s_n \mathbf{D}^n, \end{aligned} \quad (3)$$

followed by,

$$\begin{aligned} \mathbf{B}^{n+2} &= -\nabla \times \mathbf{E}^{n+1} + s_n^* \mathbf{B}^{n+1}, \\ \mathbf{D}^{n+2} &= \nabla \times \mathbf{H}^{n+1} - \sigma \mathbf{E}^{n+1} + s_n^* \mathbf{D}^{n+1} \end{aligned} \quad (4)$$

A proper choice for α_n and β_n can be used to project out eigenvectors that have eigenvalues in a specific region of the complex plane. In our formulation the trial vectors consist of the mesh wide pair (\mathbf{E}, \mathbf{B}) which are complex. The generalization also permits us to handle TEM like modes.

Previously the eigenmode analysis in MAFIA and ARGUS could be used to find the dispersion relation of a structure by modeling many periods of the structure. This led to an inordinately large number of nodes and commensurately long running times. The equivalent problem can be done using the new algorithm on a single period of the structure using an arbitrary phase shift. The number of nodes necessary to resolve a single period is greatly reduced as is the running time.

Test Problems

The new algorithm has been tested by modeling the following problems:

- A simple resonant cavity
- A cavity with a uniform resistance.
- A cavity with a jungle-gym structure
- A rectangular wave guide to derive dispersion relation
- A single period of an untapped rising-sun magnetron.

All test problems showed excellent agreement with analytical results. The lowest mode for the magnetron shows typical code output. The transverse and vertical electric field components for this problem are shown in Figure 1. The geometry of the metal structure can be seen in the figure.

Summary

We have developed a new algorithm for the extraction of complex eigenvalues and eigenvectors in the frequency-domain analysis of three-dimensional cavities. Preliminary testing shows excellent agreement with analytical results. The code is compact and CPU efficient. The code was developed and can run realistic size structures on a MacIIx with reasonable performance taking about 10-30 minutes for a typical problem. Additional testing is in progress and the results will be presented.

References

- [1] J. Tuckmated. An improved version of the eigenvector processor SAP applied in URMEL. Technical Report GERN/EF/RD 85-4, CERN, 1985.
- [2] F. Elchings, R. Klatt, E. Lawinsky, T. Weiland, S.G. Wipf, B. Steffen, T. Barts, R. K. Cooper, and G. Rodenz. Status and future of the 3D MAFIA group of codes. In Charles R. Enninger, editor, *Linear Accelerator and Beam Optics Codes*, pages 117-130, New York, 1988. American Institute of Physics. AIP Conference Proceedings No. 177.
- [3] Alan Mankofsky. Three dimensional electromagnetic particle codes and applications to accelerators. In Charles R. Enninger, editor, *Linear Accelerator and Beam Optics Codes*, pages 147-160, New York, 1988. American Institute of Physics. AIP Conference Proceedings No. 177.

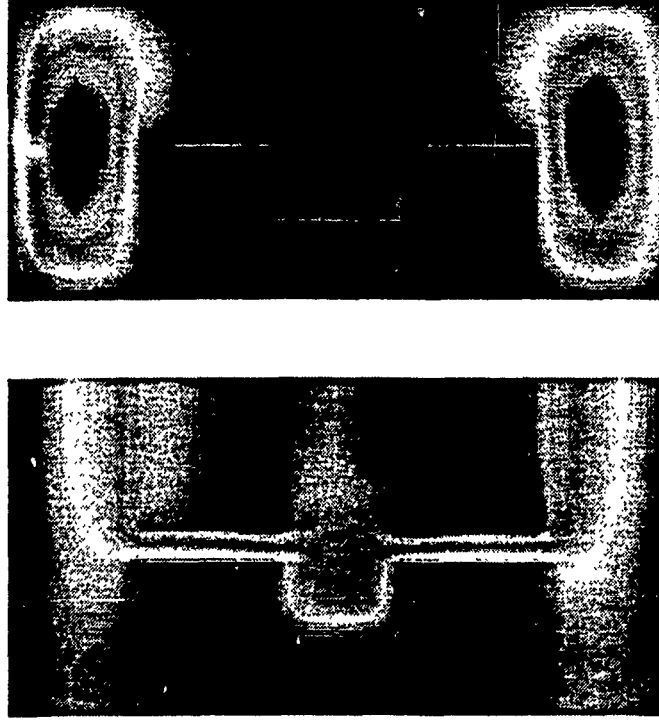


Figure 1: Electric field contours for an unwrapped rising-sun magnetron. Graph on the left is the transverse field and graph on the right is the vertical field. The black region is the metal structure, which has zero electric field.

Extension of the IFS Toroidal Particle Code (TPC) to Include Finite Beta Effects

W D Nyström and D C Barnes
Los Alamos National Laboratory
M J Lebrun and T T Tajima
Institute for Fusion Studies
University of Texas at Austin

14th International Conference on Numerical Simulation of Plasmas
Annapolis, MD, September 3-6, 1991

The IFS toroidal particle code (TPC)^{1,2} has been extended to include a self-consistent Darwin field algorithm suitable for modeling finite beta effects in toroidal fusion devices with circular cross section. Use of the Darwin algorithm removes the constraint on the time step imposed by the Courant condition for light waves present in explicit fully electromagnetic codes. A decentered Lorentz pusher is used for the electrons to remove constraints imposed on the time-step by electron gyromotion. Ions follow there full gyromotion using a standard Lorentz pusher with the leapfrog algorithm. The streamlined Darwin equations of Hewett and Boyd³ are used to facilitate solution. TPC has been written with an emphasis on portability and modularity through use of MPPL, a part of the BASIS package, and currently runs on several workstations and Cray computers.

The streamlined Darwin equations are

$$\nabla^2 \phi^n = -4\pi \rho^n \quad (1)$$

$$\mathbf{E}_L^n = -\nabla \phi^n \quad (2)$$

$$\mathbf{A}^{n+1/2} = \mathbf{A}^{n-1/2} + \mathbf{A}_L^n \quad (3)$$

$$\nabla^2 \mathbf{A}^{n+1/2} = -\frac{4\pi}{c} \mathbf{J}^{n-1/2} \quad (4)$$

$$\mathbf{B}^{n+1/2} = \nabla \times \mathbf{A}^{n+1/2} = \nabla \times \mathbf{A}^{n-1/2} \quad (5)$$

$$\zeta = \mathbf{E}_T^n - \nabla \psi^n \quad (6)$$

$$\nabla^2 \zeta^n = \frac{4\pi}{c^2} \frac{\mathbf{J}^{n+1/2} - \mathbf{J}^{n-1/2}}{\Delta t} \quad (7)$$

$$\nabla^2 \psi^n = -\nabla \cdot \zeta^n \quad (8)$$

where the subscripts L and T denote respectively the curl free (longitudinal) and divergence free (transverse) components of the appropriate vector field. At the beginning of a timestep the following information is known $\mathbf{x}^n, \rho^n, \mathbf{v}^{n-1/2}, \mathbf{J}^{n-1/2}$. The timestep begins by solving for \mathbf{E}_L^n and $\mathbf{B}^{n+1/2}$ using ρ^n and $\mathbf{J}^{n-1/2}$. The current density includes contributions from the ion electron guiding center motion and electron magnetization current. To advance

the particle velocities and positions using the leapfrog algorithm, it is necessary to know the electric and magnetic fields at timestep n. The longitudinal component of the electric field is known at the right timestep and the magnetic field can be extrapolated to timestep n using linear extrapolation. To compute \mathbf{E}_T^n using the direct method, it is necessary to know $\mathbf{J}^{n+1/2}$ which then requires knowing $\mathbf{v}^{n+1/2}$. To get $\mathbf{v}^{n+1/2}$, a predictive push of the particles is done using a predicted value of \mathbf{E}_T^n , which in the initial implementation is 0.0. The equations used to perform the predictive push of the electrons and ions are

$$\mathbf{v}_{js}^{n+1/2} = \mathbf{R}_{js}^n \cdot \mathbf{v}_{js}^{n-1/2} + \mathbf{S}_{js}^n \cdot \Delta t \frac{q_s}{m_s} (\mathbf{E}_L^n + \mathbf{E}_T^n). \quad (9)$$

$$\mathbf{R}_s = \{1 + \Theta_s^2(1 + 2\gamma_s)^2\}^{-1} \{[1 - \Theta_s^2(1 - 4\gamma_s^2)]\mathbf{I} + (1 + 2\gamma_s)2\Theta_s \cdot \Theta_s - 2\Theta_s \times \mathbf{I}\} \quad (10)$$

$$\mathbf{S}_s = \{1 + \Theta_s^2(1 + 2\gamma_s)^2\}^{-1} \{(1 + 2\gamma_s)^2 \Theta_s \cdot \Theta_s + \mathbf{I} - (1 + 2\gamma_s)\Theta_s \times \mathbf{I}\} \quad (11)$$

$$\Theta_s = \frac{\Delta t}{2} \Omega_s, \quad (12)$$

$$\Omega_s = \frac{q_s \mathbf{B}^n}{m_s c} \quad (13)$$

$$\mathbf{E}_T^n = 0 \quad (14)$$

$$\mathbf{B}^n = \frac{3}{2} \mathbf{B}^{n-1/2} - \frac{1}{2} \mathbf{B}^{n-3/2} \quad (15)$$

A predicted current density, $\mathbf{J}^{n+1/2}$ is then accumulated from the predicted velocities and used to formulate the time derivative of the current density in the field equation for the transverse electric field. The time derivative of the current density is computed as follows,

$$\left(\frac{\partial \mathbf{J}}{\partial t}\right)^n = \frac{\mathbf{J}^{n+1/2} - \mathbf{J}^{n-1/2}}{\Delta t} \quad (16)$$

$$\mathbf{J}^{n+1/2} = \sum_j q_j \sum_s \mathbf{v}_{js}^{n+1/2} S(\mathbf{x}_j - \mathbf{x}_{js}^{n+1/2}) \quad (17)$$

where

$$\mathbf{v}_{js}^{n+1/2} = \mathbf{v}_{js}^{n+1/2} + \delta \mathbf{v}_{js}^{n+1/2} \quad (18)$$

and

$$\delta \mathbf{v}_{js}^{n+1/2} = \mathbf{S}_s(\mathbf{x}_{js}^n) \cdot \Delta t \frac{q_s}{m_s} [\mathbf{E}_T^n(\mathbf{x}_{js}^n) - \mathbf{E}_T^n(\mathbf{x}_{js}^n)]. \quad (19)$$

When the source term, $\partial J/\partial t$, is formulated in this way, the equation for ξ^n takes the following form

$$\nabla^2 \xi^n - \chi^n \cdot \xi^n = \Gamma^n + \chi^n \cdot \nabla \psi^n \quad (20)$$

where Γ^n represents the part of the source term which does not depend on ξ^n and χ^n plays the same role as the susceptibility tensor familiar from implicit PIC formulations.

The field equations are all elliptic partial differential equations and are solved efficiently by Fourier transforming them in both the poloidal and toroidal angle coordinates and finite differencing in the radial coordinate. An iterative scheme is adopted for solution where the Laplacian operator is split into the cylindrical part which can be directly inverted and toroidal corrections which scale proportional to some power of the inverse aspect ratio. The toroidal correction factors for the Laplacian operator and the parts of the source terms which yield convolutions when Fourier transformed are placed on the right hand side of the field equation and the solution is iterated to convergence. For realistic inverse aspect ratios of 0.2 - 0.4, convergence is normally achieved in 4 - 16 iterations. In each case, the resulting matrix problem is tridiagonal in form with the Fourier modes being decoupled in the iterative scheme adopted so that the tridiagonal inversion can be vectorized over the poloidal and toroidal modes. Poisson's equation for the electrostatic potential yields a tridiagonal matrix system. Ampere's law for the vector potential yields a block tridiagonal matrix with 2 by 2 blocks for the radial and poloidal components and a tridiagonal matrix for the toroidal component. The coupled set of equations for the transverse electric field yields a block tridiagonal matrix with 4 by 4 blocks of complex coefficients.

This extended version of TPC is currently being tested and validated. Results of these tests will be presented as well as performance comparisons for runs on various computational platforms.

References

- 1 M. J. Lebrun, University of Texas Report IFSR # 316 (1988).
- 2 M. J. Lebrun and T. Tajima, University of Texas Report IFSR # 374 (1989).
- 3 D. W. Hewett and J. K. Boyd, "Streamlined Darwin Solution of Nonneutral Plasmas," J. Comp. Phys. 70, 166 (1987).

Computational Method for Deterministic Chaos in a Magnetically Confined Plasma

Erk L. Vold
Applied Theoretical Physics Division, F664
Los Alamos National Laboratory
Los Alamos, NM 87545

Deterministic chaos, originally seen in the Lorenz equations, designates the concept that turbulent behavior arises from systems of non-linear equations, including the fluid moment or Navier-Stokes equations, and specifically the fluid approximation to a magnetized plasma. Consistent computations capable of reproducing 'anomalous' transport observed in tokamaks are assumed to rely upon the correct computational treatment of the dominant macroscopic nonlinearities in the x - y plane. These nonlinear effects include the divergence of the momentum flux tensor in magnetic flux surface coordinates, the core and edge plasma treated as a single region coupled to the sheath and edge physics boundary conditions including neutral particle recycling, and the plasma potential, ϕ , calculated in non-ambipolar conditions. Unresolved subgrid transport and plasma kinetic effects can be added in the momentum viscosity tensor. An iterative and time dependent solution for the coupled quantities: n , e , T_e , and ϕ , J , T_e , with B and diffusive neutrals and impurity species is described for the toroidally symmetric case in (θ, ψ, ϕ) coordinates. An inner iteration solves the transport problem in an estimated flux surface geometry assuming $E = -\nabla\psi \times \hat{\phi} + E_{\parallel} \hat{\phi}$, and an outer iteration couples the plasma transport to consistent MHD surfaces. The control volume form of the momentum flux tensor in magnetic flux surface geometry, an elliptic equation for ϕ , and algorithm sweep considerations for the magnetic surface topology are emphasized.

The concept of deterministic chaos was introduced by Lorenz [1], in his landmark study of chaotic behavior in the solution of a system of three coupled non-linear ordinary differential equations representing a Rayleigh-Bénard type of instability. Chaotic solutions have been observed specifically in a plasma, representing turbulent behavior in the Hasegawa-Mima model for drift waves [2]. A similar drift wave model with an added poloidally periodic forcing function leads to chaotic or turbulent like solutions [3]. Such a poloidal variation could be driven by asymmetries in the plasma edge region.

Global confinement is limited by anomalous transport in the tokamak generally regarded to be driven by electrostatic fluctuations observed to be most pronounced in the plasma edge region [4]. The edge plasma is also critical in H-mode confinement. Speculation relates the steady state and the fluctuating plasma potential to the anomalous transport mechanism [5], and suggests that the poloidal asymmetries driven in the edge plasma by flow to the target sheath coupled with the strong poloidal drift flows are sufficient to generate the observed edge plasma turbulence [6].

The turbulence observed in the plasma edge is characterized as broad-band low frequency, so that computations with time steps on the order of 10^{-5} - 10^{-6} seconds resolve the frequency spectrum in which most of the fluctuation power is observed. The turbulent spatial scale length spectrum is observed to peak near $k \rho_i = 0.1$ (for ion gyroradius, ρ_i). This corresponds for typical edge plasma values to scale lengths of 0.3 - 1.0 cm, which is the same range expected for the Scrape-Off-Layer (SOL) radial decay in plasma density (or e-folding length). This range will be resolved by the computational grid in the radial direction but may require a poloidal subgrid model. Toroidal fluctuations for $m > l$ are unresolved in this two dimensional calculation.

These introductory considerations lead us to expect to resolve tokamak turbulent transport provided that the dominant non-linearities are retained. These non-linear effects include: the divergence of the momentum flux tensor in magnetic flux surface coordinates, the core and edge plasma treated as a single region coupled to the sheath and edge physics boundary conditions including neutral particle recycling, and the plasma potential, ϕ , calculated in non-ambipolar conditions.

An iterative and time dependent solution algorithm is shown in Table 1 for the coupled quantities: n , e , T_e , and ϕ , J , T_e with B determining the flux surface grid geometry and diffusive neutrals and impurity species for the toroidally symmetric case. An inner iteration solves the transport problem in an estimated flux surface geometry assuming $E = -\nabla\psi \times \hat{\phi} + E_{\parallel} \hat{\phi}$ and an outer iteration (to be added in a later phase of the project) couples the plasma transport to consistent MHD surfaces, and includes the inductive components of the electric field. Here, we assume that the estimated magnetic flux surface is adequate to treat the transport in a tokamak dominated by electrostatic forces.

The magnetic flux surface geometry which couples the core plasma to the edge plasma is shown in Fig. 1a. This is a general up-down asymmetric case showing four magnetic flux surfaces: a closed core plasma surface, ψ_1 , the flux surface of the lower x -point, ψ_2 , the flux surface of the upper x -point, ψ_3 , and a flux surface which is outside both x -points, ψ_4 . Large poloidal asymmetries are observed in computations of the edge plasma region [6], while they are assumed to be negligible in most core plasma transport computations, further motivating the consistent core-edge plasma coupling.

For computational efficiency using regular coordinate indices, the actual geometry in Fig. 1a is mapped into a topologically rectangular region as shown in Fig. 1b. The two x -points are stretched into the top and bottom x -point lines. The horizontal line down the center of Fig. 1b represents three singular points in the real geometry, including the core plasma center point. The left and the right halves of Fig. 1b are unrelated across this central horizontal line, so solution in the horizontal direction (across the magnetic flux surfaces) involves two implicit sweeps. The figure shows the fluxes across the x -point lines which must be matched in the computational boundary conditions to retain the original geometry and match flows through the core, the edge and the two private flux regions. This reduces to six separate computational regions for implicit sweeps when solving in the vertical direction (along the magnetic flux surface). One implicit step then involves a vertical sweep through six regions and a horizontal sweep through two regions. The non-linearity of the problem requires at least a second implicit step in each iteration or time increment.

The momentum equation is solved in the form:

$$\frac{\partial \rho u}{\partial t} + \nabla \cdot (\rho u u) + \nabla p + \nabla \cdot \Pi = J \times B + S_p$$

and discretized using a control volume formulation on a staggered grid. The viscosity tensor, Π , can be used to include classical viscosity, kinetic effects omitted in the fluid formulation of the plasma transport problem or anomalous viscosity effects. It is assumed in the initial phase, that the desired transport results from the advection term and that the viscous tensor is negligible.

The control volume form of the momentum flux tensor in magnetic surface geometry (θ, ψ, ϕ) is critical to represent the tokamak geometry accurately. It is written for the (θ, ψ, ϕ) components with toroidal symmetry as:

$$\int (\nabla \cdot \rho u u)_{\theta} dV = \int \partial_{\theta} (\Lambda_{\theta\theta} u_{\theta} u_{\theta}) + \int \partial_{\psi} (\Lambda_{\theta\psi} u_{\theta} u_{\psi}) + u_{\theta} u_{\psi} \int \partial_{\psi} (L_{\theta}) - u_{\psi} u_{\theta} \int \partial_{\theta} (L_{\psi})$$

$$\int (\nabla \cdot \rho u u)_{\psi} dV = \int \partial_{\theta} (\Lambda_{\psi\theta} u_{\psi} u_{\theta}) + \int \partial_{\psi} (\Lambda_{\psi\psi} u_{\psi} u_{\psi}) + u_{\psi} u_{\psi} \int \partial_{\psi} (L_{\psi}) - u_{\psi} u_{\theta} \int \partial_{\theta} (L_{\psi})$$

$$\int (\nabla \cdot \rho u u)_{\phi} dV = \int \partial_{\theta} (\Lambda_{\phi\theta} u_{\phi} u_{\theta}) + \int \partial_{\psi} (\Lambda_{\phi\psi} u_{\phi} u_{\psi}) + u_{\phi} u_{\psi} \int \partial_{\psi} (L_{\phi}) + u_{\psi} u_{\phi} \int \partial_{\theta} (L_{\phi})$$

where L denotes a computational cell length in the subscripted direction and (δl) denotes a finite difference of the bracketed quantity in the subscripted direction. The terms associated

with the curvature of the magnetic flux surfaces (the second line of each of the component equations above) includes centrifugal forces and flux expansion terms significant in divertor geometries. The toroidal momentum is required to consistently close the equation set.

Non-ambipolar flows are incorporated by solving Ohm's laws for currents:

$$\nabla \phi + \eta \mathbf{j} + \frac{1}{n} \frac{\mathbf{x} \cdot \nabla}{r} = \frac{\nabla A}{n} + \frac{\nabla T_e}{n} + \mathbf{u} \times \mathbf{B} + d\mathbf{P} + \mathbf{E}_I$$

where $d\mathbf{P}$ is a general source term to account for externally applied sources or neutral-plasma interactions, and \mathbf{E}_I is the inductive part of the electric field. The plasma potential is solved by rearranging the equation, $\nabla^2 \phi = 0$, as an elliptic equation to solve for ϕ [6]. This can be written in a coordinate specific form so that it is not directly dependent upon the current, but only upon the plasma profiles which are assumed or known at the time of the plasma potential computation.

Plasma ion and electron temperatures are solved from the standard energy equation for non-ambipolar flow [6]. Classical thermal conduction includes the drift direction contribution proportional to (T_e/B) , and can be significant for non-zero poloidal gradients. Neutral transport is solved on the same computational grid in a discretization of a fluid diffusion approximation. A one-group model in thermal equilibrium locally with the ions [7] is proposed, and later, an improved multi-group formulation [8] is envisioned. The neutral model provides the essential coupling to the plasma source terms arising from the particle recycling in the edge plasma regions. Similar diffusion based models are proposed for an impurity species to obtain the important modification to the energy balance arising from radiative losses in the presence of the higher atomic number impurities.

Boundary conditions for each of the plasma transport equations are related to the Bohm sheath condition setting the parallel flow to the plasma sonic speed, C_s . This fixes the poloidal speed to the target plates at $(B\theta/B) C_s$, and fixes the poloidal flux for the plasma momentum and density. Non-ambipolarity prescribes the current and plasma potential boundary conditions. Energy loss to the sheath is related to standard sheath transmission factors for the electron and the ion power.

In conclusion, a significant amount of plasma density, momentum and heat are lost to the sheath boundary conditions in the plasma edge region with an equivalent flux driven across the separatrix by some mechanism. We assume this flux corresponds to the observed anomalous transport, and that the mechanism is nonlinear advection. This advection can be resolved by coupling the three components of plasma fluid momentum and currents consistently in the tokamak geometry. If advection is proven insufficient to account for the observed flux, then a small value of anomalous viscosity can be introduced to match the experimental results.

Acknowledgement - This work was performed in the Post-Doctoral Fellowship Program at the Los Alamos National Laboratory operated for the U.S. Department of Energy by the University of California. The author thanks D. C. Barnes for helpful discussions.

References

- [1] E.N. Lorentz, *J. Atmos. Sci.*, **20**, 130 (1963).
- [2] Hasegawa and Mima, *Phys. Fluids*, **21**, 87 (1978).
- [3] Kallifien and Salat, *Plasma Phys. & Contr. Fusion*, **31**, 123 (1989).
- [4] TTF sum in *Phys. Fluids B* 777
- [5] TTF sum or Stangeby review in *Nucl. Fusion* 777
- [6] Vold, Najmabadi, and Conn, accepted in *Phys. Fluids B*, MS PF-18700 (1991).
- [7] Vold, Prinja, Najmabadi, Conn, *J. Nucl. Mater.*, **176&177**, 570 (1990).
- [8] Hasan and Conn, *J. Comput. Phys.*, **71**, 371 (1987).

Table 1. Paper Tokamak Main Computational Flow

1. Initial plasma profiles, sources and sinks assumed. Flux Surface Coordinates (θ, ψ, ϕ) mapped into (R, Z) geometry. Initial topologically rectangular grid indexing assignment (regular ij increments).
2. Set dt , time step solutions and iteration options specified.
3. Magnetic surfaces
4. Toroidal plasma currents, $j_\theta(\theta, \psi)$, mapped into (R, Z) geometry.
5. Poloidal flux function, ψ , evaluated.
6. $B_\theta(R, Z)$ and $B_z(R, Z)$ computed (from ψ or from j_θ).
7. Inductive components of \mathbf{E} determined.
8. Flux surface geometry calculated, and mapped to topologically rectangular grid (optionally re-map plasma solution to new volumes)
9. Plasma transport computed in (θ, ψ, ϕ) Coordinates:
 - Plasma sources and sink terms
 - External sources specified
 - Neutral density, n_0 transported by diffusion
 - Impurity density, P_{imp} transported by mean flow advection and species specific diffusion
 - Plasma (ion) equations:
 - Momentum (θ, ψ, ϕ) equations
 - Continuity
 - Ion temperature or energy equation
 - Non-ambipolar (electron) equations
 - Plasma potential
 - Plasma currents (θ, ψ, ϕ)
 - Electron temperature or energy equation
10. Relax plasma transport (go to 8), iterating on $(\rho, u, T_e, \phi, j, T_e, P_{imp})$
11. Relax transport: magnetic flux surface coupling (go to 3)
12. Advance to next time step (go to 2).

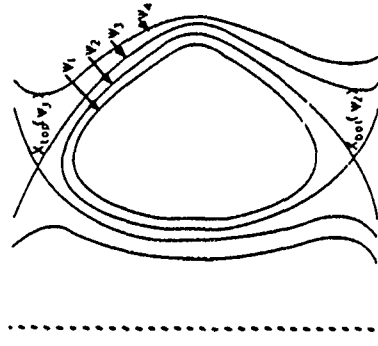


Fig. 1a. General divertor geometry with up-down asymmetry and x-points on different flux surfaces. Fig. 1b. Topologically equivalent rectangle for the computational grid

Numerical Simulation of Resonant Alfvén Wave Absorption

Robert Holdaway and Robert A. Smith
Department of Physics and Astronomy, The University of Iowa, Iowa City IA 52242

James L. Geary
Berkeley Research Associates, Springfield VA 22150

Resonant Alfvén wave absorption occurs when a magnetohydrodynamic (MHD) wave propagates into an inhomogeneous region containing one or more resonance layers. At the resonance layers the local Alfvén speed matches the phase speed of the wave. At the resonance layer the incident wave is mode-converted into a kinetic Alfvén wave which is Landau damped. For low-beta resonance the electrons are accelerated. Ions are accelerated for beta close to unity. Characteristics of the absorption process are studied with a 2 1/2-D magnetoinductive (Darwin) particle simulation employing guiding-center electron dynamics and fully-kinetic ions. The code includes both gradient and curvature effects to study inhomogeneities on the scale of the ion gyro-radius.

Model

The time-scale of Alfvén wave absorption is much longer than the electron gyro-frequency ($\omega \ll \Omega_e$) and the spatial scale much longer than the electron gyro-radius ($k\rho_e \ll 1$). For this reason the electron motion is treated in the guiding-center model

Electrons

$$\begin{aligned}\frac{dv_{e\parallel}}{dt} &= \frac{e}{m} E_{\parallel} - \frac{\hbar c}{m} \frac{\nabla B_{\parallel}}{B} + (v_{e\parallel}^2 (\mathbf{b} \cdot \nabla) \mathbf{b})_{\parallel} \\ v_{e\perp} &= \frac{c \mathbf{E} \times \mathbf{b}}{B} + \frac{\mathbf{b}}{\Omega_e} \times \left(v_{e\parallel}^2 (\mathbf{b} \cdot \nabla) \mathbf{b} + \frac{\mu \nabla B}{m} \right) \\ \frac{dx_e}{dt} &= v_{e\perp} + v_{e\parallel} \mathbf{b}\end{aligned}$$

Ions

$$\begin{aligned}\frac{dv_i}{dt} &= \frac{e}{M} \mathbf{E} - \frac{e}{Mc} \mathbf{v}_i \times \mathbf{B} \\ \frac{dx_i}{dt} &= \mathbf{v}_i\end{aligned}$$

(note: For this code the electrons carry a positive charge and the ions a negative charge.)

The Darwin model neglects the transverse component of the displacement current, excluding radiation. Magnetization current is included.

$$\nabla \cdot \mathbf{E}_L = 4\pi p$$

$$\nabla \times \mathbf{B} = \frac{4\pi}{c} (\mathbf{J}_i + \mathbf{J}_{GCe} - c \nabla \times \langle \mu_e \rangle \mathbf{b} + \mathbf{J}_{exDT})$$

$$\nabla^2 E_T = \frac{4\pi}{c^2} \frac{\partial J_T}{\partial t}$$

$$\begin{aligned}\frac{\partial \mathbf{J}}{\partial t} &= \frac{\partial \mathbf{J}_{exDT}}{\partial t} + c \nabla \cdot \langle \mathbf{v} \mathbf{v} \rangle_{\perp} + \frac{e^2}{M n_i} \mathbf{E} - \frac{e}{Mc} \mathbf{J}_i \times \mathbf{B} \\ &\quad - c \nabla \cdot \langle \mathbf{v} \mathbf{v} \rangle_e + \frac{e^2}{M n_e} \mathbf{E}_\parallel - c \frac{\langle \mu_e \rangle \nabla B_{\parallel}}{m} \mathbf{b} + c \langle v_{e\parallel}^2 \rangle (\mathbf{b} \cdot \nabla) \mathbf{b}\end{aligned}$$

$$\nabla \cdot \mathbf{E}_T = \nabla \times \mathbf{E}_L = 0$$

The fields are calculated on the grid using the Fast Fourier Transform (FFT). Charges and currents are accumulated on the grid using SUDS weighting. A gaussian particle shape is attributed to the particles as the fields are calculated in k-space. Calculation of the transverse electric field is accomplished by iteration.

The simulation boundary conditions are doubly periodic; each simulation run treats two different non-uniform profile physics problems. A sheet antenna is positioned with its wave-vector along y at the center of the simulation in a region of uniform plasma. The y-direction contains a component of the background magnetic field resulting in a k_y . The compressive Alfvén wave (CAW) propagates away from the antenna in the x-direction (perpendicular to the background magnetic field) into non-uniform regions where the resonances are located. Near the x-boundaries the plasma is again uniform.

The result of our runs is heating at the resonance layers. We find that the heating rate depends on the non-uniform profile's scale-length. The longer scale-lengths accelerate electrons along the magnetic field more quickly and to greater velocities. The ions are not heated along the magnetic field for low-beta. The results of several runs will be shown in the poster session.

Porting A Global Ocean Model Onto A Shared-Memory Multiprocessor: Observations and Guidelines

Richard J. Procassini, Scott R. Whitman and William P. Dannevik
Lawrence Livermore National Laboratory
University of California
Livermore, CA 94550

Global climate modeling is one of the "grand challenge" applications that will benefit from the predicted increase in computing performance associated with the use of massively parallel computers. An accurate prediction of our planet's climate requires a complete model which couples an atmospheric and an oceanic description of circulation, as well as heat and moisture transfer processes. The small space and large time scales associated with oceanic flow suggests that the ocean portion of the calculation may be the most computationally intensive. For this reason, it has been decided to use a global-ocean general circulation model (GCM) as a testbed with which to study issues related to the parallelization of climate models on multiple-instruction stream, multiple-data stream (MIMD) parallel computers.

The Semtner and Chervin global ocean GCM [1, 2] has been modified to run on the BBN TC2000 "Butterfly" MIMD computer [3]. This implementation of the three-dimensional primitive-equation GCM employs a shared-memory scheme for data management which is similar to the previous use of solid-state disk (SSD) on Cray computers. The ocean GCM has been ported to run on the TC2000 under both the BBN TC2000 Fortran (TCF) [4] and Parallel Fortran Preprocessor (PFP) [5] programming models. The use of shared-memory constructs for data management introduces a number of factors which can degrade the performance of the code in a massively-parallel computing environment. In an attempt to characterize the parallel computing performance of this shared-memory version of the

code, it has been run on the TC2000 with processor counts in the range of $1 \leq P \leq 60$. A detailed analysis of these timing results has been performed to determine the impact of several of the performance degradation factors on the parallel computing performance of the shared-memory global ocean model.

Acknowledgments

The authors wish to thank Karen Warren, Tammy Welcome and Brent Gorda (Massively Parallel Computing Initiative), and Bill Celmaster (BBN Advanced Computers) for helpful discussions and tutorials on use of the Uniform Systems and PFP programming models. This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under contract number W-7405-ENG-48.

References

- [1] A.J. Semtner, "Finite-Difference Formulation of a World Ocean Model", in *Advanced Physical Oceanographic Numerical Modelling*, (Reidel Publishing Company, Dordrecht, 1986), pp. 187 - 202.
- [2] A.J. Semtner and R.M. Chervin, *J. Geophys. Res.*, **93**, 15502 (1988).
- [3] M. Beeler, "Inside the TC2000 Computer, First Release", (BBN Advanced Computers Incorporated, Cambridge, MA, 1990), pp. 59 - 62.
- [4] A. Quigley, "TC2000 FORTRAN Reference, Revision 2.0", (BBN Advanced Computers Incorporated, Cambridge, MA, 1990), pp. 331 - 384.
- [5] K. Warren, B. Gorda and E.D. Brooks III, "Programming in PFP", Report UCLL-MA-107028, Lawrence Livermore National Laboratory, Livermore, CA, 1991.

Wednesday Evening
September 4, 1991

Poster Session (PWE)
8:00pm - 10:00pm

Regatta Ballroom C

Graphical Display and Animation of Data Produced by Electromagnetic, Particle-In-Cell Codes

M. L. Kiefer, T. D. Poinson, D. B. Seidel,
R. S. Coats, and J. P. Quintenz
Sandia National Laboratories
Albuquerque, New Mexico

OVERVIEW.

Charged-particle simulations are performed routinely in the Pulsed Power Sciences Directorate at Sandia with the three-dimensional, finite-difference, electromagnetic, particle-in-cell (EM-PIC) QUICKSILVER^{1,2} code. TWOQUICK, a two-dimensional version of QUICKSILVER, is currently under development. Areas of simulation with these codes include ion diodes, magnetically insulated transmission lines, microwave devices, and electron beam propagation. The computing philosophy used to design these codes dictates that there are no calls to graphics routines in the simulation codes themselves; rather, these codes, which run on supercomputers, produce data files for analysis (postprocessing) on computers, such as high-performance graphics workstations, connected to the supercomputer via a network. This design philosophy allows the simulation codes and the postprocessing tools to be used on the hardware platform most appropriate for the tasks. The generic postprocessing tasks involve both quantitative and qualitative analysis of the data produced by these codes using graphics and animation. The PLOTFFF code, developed at Sandia, is used for the quantitative postprocessing tasks. Stander's Application Visualization System^{3,4} software (AVS), which uses graphics produced by the OS/VS code developed at Sandia, is used for the qualitative postprocessing tasks. Examples of the capabilities of PLOTFFF and AVS/QSAVS will be shown. A videotape of animations will also be presented.

DATA FILE FORMAT:

The transfer of the 1-yr data files generated on a Cray supercomputer to a remote computer, such as a UNIX workstation or VAX computer, for postprocessing requires a file format that is portable among machines with vastly different word structures and that is minimal in size. It is quite easy to generate millions of 64-bit words of data from a QUICKSILVER job that must be postprocessed. To address this, the Portable File Format⁵ (PFF) and supporting library of modules has been developed at Sandia. The PFF module library consists of low and high level routines which write data into a file (and of course, read it back) in a format which is easily passed between machines. Typically, only a few of the low level routines must be modified when porting to new machines. The high level routines, which are the only ones directly called by a user writing an application, are completely portable. Both FORTRAN and C versions of the PFF library are available. Using the library, all data are written to the PFF file as two-byte integers. The format of a PFF file is quite general and can support a wide variety of applica-

tions. The PFF format is also extensible to accommodate almost any data type desired. Both QUICKSILVER and TWOQUICK write PFF files and both PLOTFFF and QSAVS read PFF files. The use of the PFF file format, in addition to minimizing the difficulty of transferring data between computers, allows a wider use of postprocessing tools. Any simulation code which writes PFF files could use PLOTFFF and AVS/QSAVS.

GENERIC DESCRIPTION OF FINITE-DIFFERENCE EM-PIC DATA:

We describe data from these codes as being either "grid-data", "particle-data", "surface-data" or "time-histories". Grid-data are data that lie on the finite-difference grid and have some number of attributes for each grid-point. For example, scalar data, such as charge density, has 1 attribute and vector data, such as electric field, has 3 attributes. Particle-data are data that are associated with the unordered, charged particles in the simulation and, again, have some number of attributes for each particle. Particle positions have no attributes, particle charges have one attribute and particle momenta have 3 attributes. Surface-data are complex surfaces which are conformal with the grid and contain surface-normal information and may contain attributes. For example, surfaces representing conducting boundaries have no attributes and magnetic field on a conductor would have 2 attributes. We denote a "snapshot" of grid-, particle- or surface-data in a PFF file as being a single set of data, e.g., charge density, at either a single point in time or averaged over a small number of time-steps. These snapshots may lie in two or three dimensional spaces and the space may be Cartesian, cylindrical or spherical. Time-histories are just a single data value at a sequence of time-points. A snapshot or a time-history is denoted generically as a single "dataset" in a PFF file. A PFF file may contain many datasets, all of which may be of different types; additionally, QUICKSILVER and TWOQUICK can write multiple PFF files.

PLOTFFF POSTPROCESSOR:

The PLOTFFF postprocessor is used for quantitative analysis of these EM-PIC datasets. It is written in ANSI-standard FORTRAN and includes a portable graphics library for using a variety of graphics devices. It runs on VAX computers and UNIX workstations with X Windows. PLOTFFF can utilize multiple PFF files simultaneously.

Snapshots or time-histories can be manipulated and displayed with PLOTFFF. A single data attribute of grid-data snapshots, such as scalar data, one vector component or vector magnitude, can be extracted by PLOTFFF and two-dimensional slices of that data can be displayed as color-contour plots. Also, one-dimensional pencils of the data can be displayed as x-y plots, which are denoted as "line-outs". PLOTFFF also can apply Fourier transforms to these data and integrate the data before display.

PLOTFFF can manipulate particle-data snapshots to produce scatter plots of particles on phase-space planes or color contours of phase-space density. In addition, particle trajectory plotting capability is planned.

Time-histories can be displayed as line-outs. Any type and number of line-outs can be displayed and overlaid using PLOTFFF.

PLOTPFF can overlay conductor boundary surfaces which intersect a two-dimensional slice of grid-data or particle-data with the contour or scatter plots of that slice. Such intersections appear as a set of line segments and help to orient the user.

AVS AND QSAVS POSTPROCESSORS:

AVS and QSAVS are used for the qualitative postprocessing tasks by applying three-dimensional graphics and animation techniques. AVS is built upon the PHIGS ANSI-standard graphics package, the proposed ANSI-standard PHIGS+ extensions, and the market-standard X Windows software, so it is fairly portable and is available on a variety of UNIX workstations. It is also easily extensible by the user for incorporation of specific graphics methods not available in the basic software and for incorporation of routines to read user-specific input files. Our primary use of AVS is to interactively manipulate and animate geometric objects, e.g., a set of spheres representing particle positions, using the AVS Geometry Viewer. QSAVS takes time sequences of snapshots and produces three-dimensional geometric objects which are then input to the AVS Geometry Viewer for manipulation and animation. QSAVS automates the production of animations and the unattended video-recording of those animations. It is written in C and currently only runs on a Stardent workstation. As AVS becomes more widely available, it should be easy to port QSAVS to other vendors' workstations. A version for Hewlett Packard systems should be available within 6 months.

We can use AVS to produce various graphics objects: spheres, line segments, pyramids (approximate cones), and surfaces. Spheres consist of location of the sphere centers (or vertices), their radii, and, possibly, their color. Line segments consist only of endpoints vertices and, possibly, the color at each vertex. Pyramids consist of the location of each vertex, the surface normal at each vertex, and, possibly, the color of each vertex. Surfaces consist of a number of vertices describing the surface, normals at each vertex, and, possibly, the color at each vertex. The Geometry Viewer takes these primitive objects and allows for interactive rotation, translation, and scaling of the objects. Color and transparency of objects can also be manipulated interactively with the Geometry Viewer. Multiple views of objects can be performed and primitive objects can be grouped into compound objects so that either the primitive or the compound object can be manipulated. Finally, AVS can display real-time animation of these objects, typically at a rate of 3-6 frames per second, where one frame may have an object with 1,000 to 10,000 vertices. An entire animation often consists of as many as 500 frames. The post-processing, then, becomes the task of mapping EM-PIC data onto these three-dimensional graphics primitives. This is the function provided by QSAVS. Methods for mapping data attributes onto attributes of these graphical primitives is described below.

Particle positions, defined as particle-data with no attributes, can be displayed with constant radius spheres centered at the location of each particle. Different particle species, such as electrons and protons, can be differentiated with the use of color. A three-dimensional field of scalar data, single vector components or vector magnitudes in a grid-data or particle-data snapshot can also be displayed using spheres centered at the location of each data-point. In this case, the data attribute can be mapped to the hue-based color of each sphere. For a two-dimensional slice of grid-data, single data attributes can be mapped to the hue of each vertex of a planar surface conformal to the

slice and then AVS smoothly interpolates the color at every pixel on the surface to produce an extension of a classical contour plot. Vector grid-data and particle-data snapshots (3 data attributes) can be displayed in a variety of ways. One way is to use line segments which have their first endpoints located at each of the data-points. The orientation and color of a segment will correspond to the direction and magnitude, respectively, of the vector. Another method of viewing vector data is to use pyramids which have their bases centered at each data point. Again, the orientation and color of a pyramid will correspond to the direction and magnitude, respectively.

QSAVS provides the capability to reduce the range of single data attributes (clamping) and to remove data points outside of a given range (thresholding). It allows for duplicating data across planes of mirror and periodic symmetry. Systematic removal of data points in a snapshot (sparsification) or removal of data attributes (for display of vector components) can also be performed.

VIDEOTAPING ANIMATIONS:

We have the capability to convert the high-bandwidth video from the workstation to NTSC video formats, either composite, Y/C, or RGB, using a scan converter. We use tools for unattended, frame-by-frame recording of the graphical animation, displayed using AVS, onto video laser disk units. We then utilize programmed playback of the animation on the laser disk for dubbing onto videotape. For instance, frames on the laser disk can be played in any order, at any speed (including pauses), in either forward or reverse directions.

1. D. B. Seidel, M. L. Kiefer, R. S. Coats, A. L. Siegel, J. P. Quintenz, Proc. 12th Conference on Numerical Simulation of Plasmas, San Francisco, CA, paper PT-24 (1987).
2. M. L. Kiefer, D. B. Seidel, R. S. Coats, J. P. Quintenz, T. D. Pointon, and W. A. Johnson, Proc. Conference on Codes and the Linear Accelerator Community, Los Alamos, NM (1990).
3. C. Upson, T. Faulhaber, D. Kamins, D. Laidlaw, D. Schlegel, J. Vroom, R. Gurwitz, and F. Moss, IEEE Computer Graphics and Applications, "The Application Visualization System: A Computational Environment for Scientific Visualization", (July 1989).
4. Application Visualization System, Volumes 1 and 2, (User's Guide, Module Reference and Developer's Guide), Stardent Computer Inc., Concord, MA (1991).
5. D. B. Seidel, R. S. Coats, M. L. Kiefer, T. D. Pointon, and L. P. Mix, 9th Biennial CUBE Symposium, Santa Fe, NM (1990).

• This work was supported by the U.S. Department of Energy under Contract No. DE-AC04-76DP00789

A Post-Processor for the PEST code
S. Preuschel, J. L. Johnson, J. Mauckam
Princeton University

A new post processor for use with output from the PEST code¹ has been developed. It allows us to use quantities calculated by PEST and take better advantage of the physics picture of the plasma instability which they can provide. This will improve comparison with experimentally measured quantities as well as facilitate understanding of theoretical studies.

The PEST code is used to determine the linear ideal MHD stability of axisymmetric tokamak configurations. It is a variational code which determines the set of displacement vectors ξ which minimize the Lagrangian, $\delta W - \omega^2 K = 0$, where δW and K are the potential and kinetic energies respectively associated with perturbations from a given equilibrium. In practice only the most unstable eigenvector and its corresponding eigenvalue are determined. The mode structure of the perturbed quantities gives us some clues into the physical origin of the instability. The PEST code has not fully exploited this capability. In fact even the usual plots of the Fourier components of ξ are misleading since they do not include the proper normalizations. To remedy this deficiency using the new post processor we may now

- construct properly normalized components of the eigenfunction ξ ,
- look at the eigenfunction in an orthogonal basis, as opposed to PEST's non-orthogonal basis,
- compute quantities which can be measured experimentally, e_θ components of the perturbed magnetic field Q ,
- test the importance of compressibility for a specific eigenvalue by evaluating $\nabla \cdot \xi$.

For a given tokamak plasma equilibrium, the PEST code determines ξ_θ^P , ξ_r^P , ξ_ϕ^P , where the displacement vector ξ for perturbations about the equilibrium is

$$\xi = \frac{J \xi_\theta^P}{g R^2} \nabla \theta \times B + \frac{i J \xi_r^P}{g R^2} B \times \nabla \psi + \kappa_B^P B \quad (1)$$

Each component, α , of ξ has been decomposed such that

$$\xi_\alpha^P = \sum_{m,n} \xi_{\alpha,mn}(\psi) e^{i(m\theta - n\phi)} \quad (2)$$

Since axisymmetry is assumed, the Fourier coefficients for different values of n decouple so that each toroidal mode number can be examined separately. However, the magnitudes of the PEST coordinates $\nabla \psi$, $\nabla \theta$, $\nabla \phi$, and the Jacobian, $J = \nu \lambda^2 / 2\pi R$, have θ dependencies so that graphs of these Fourier components do not describe the actual eigenfunctions. To get a better physical picture of the modes of the displacement vector, we define a set of unit vectors,

$$\hat{e}_\psi \equiv \frac{\nabla \theta \times B}{|\nabla \theta \times B|}, \quad \hat{e}_\theta \equiv \frac{B \times \nabla \psi}{B |\nabla \psi|}, \quad \hat{e}_\phi \equiv \frac{B}{B}. \quad (3)$$

where

$$\begin{aligned} \xi &= \xi_\psi \hat{e}_\psi + \xi_\theta \hat{e}_\theta + \xi_\phi \hat{e}_\phi \\ &= \frac{\nu \lambda}{2\pi g R^2} \left[R^2 g^2 |\nabla \theta|^2 + f^2 (\nabla \psi \cdot \nabla \theta)^2 \right]^{1/2} \xi_\psi^P \hat{e}_\psi \\ &\quad + i \frac{\nu \lambda |\nabla \psi|}{2\pi g R^2} \left[R^2 g^2 + f^2 |\nabla \psi|^2 \right]^{1/2} \xi_r^P \hat{e}_\theta \\ &\quad + \frac{i}{\lambda} \left[R^2 g^2 + f^2 |\nabla \psi|^2 \right]^{1/2} \xi_\phi^P \hat{e}_\phi, \end{aligned} \quad (4)$$

and Fourier decompose ξ_ψ , ξ_θ , ξ_ϕ in θ . Shown in figure 1 are the original ξ_θ^P and the new ξ_θ mode structures for a TFTR type plasma with $\beta > 4\%$. This mode is pressure driven at the center and current driven at the edge. With the proper normalization we see a difference in the relative magnitudes of the modes as well as sign changes in the modes and the appearance of the $m=0$ mode.

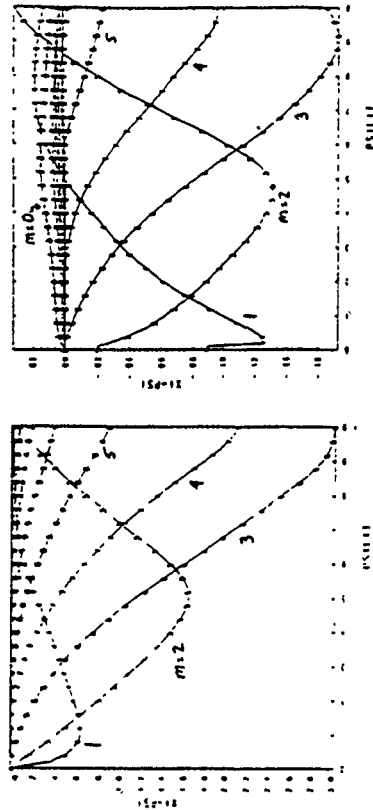


Figure 1

ξ_θ after normalization

Since the PEST basis vectors are not orthogonal, i.e. $\nabla \psi \cdot \nabla \theta \neq 0$, it is useful to look at ξ in an orthogonal system. Thus we define a new set of orthogonal unit vectors

$$\hat{e}_r \equiv \frac{\nabla \psi}{|\nabla \psi|}, \quad \hat{e}_\theta \equiv \frac{B \times \nabla \psi}{B |\nabla \psi|}, \quad \hat{e}_\phi \equiv \frac{B}{B}. \quad (5)$$

Then we can write,

$$\begin{aligned} \xi &= \xi_\psi \hat{e}_\psi + \xi_\theta \hat{e}_\theta + \xi_\phi \hat{e}_\phi \\ &= \frac{\xi_\psi^P}{R |\nabla \psi|} \hat{e}_r \\ &\quad + \frac{\nu \lambda |\nabla \psi|}{2\pi g R^2} \left[R^2 g^2 + f^2 |\nabla \psi|^2 \right]^{1/2} \left(\xi_\theta^P - \frac{(\nabla \psi \cdot \nabla \theta) \xi_\phi^P}{|\nabla \psi|^2} \right) \hat{e}_\theta \\ &\quad + \frac{i}{\lambda} \left[R^2 g^2 + f^2 |\nabla \psi|^2 \right]^{1/2} \xi_\phi^P \hat{e}_\phi \end{aligned} \quad (6)$$

This form is useful for looking at radial displacements, as \hat{e}_r is always perpendicular to a flux surface, whereas \hat{e}_θ has components in all three coordinate directions, i.e.,

$$\nabla \theta \times B = \frac{R g J |\nabla \theta|^2}{\lambda^2} \nabla \psi - \frac{R g J (\nabla \psi \cdot \nabla \theta)}{\lambda^2} \nabla \theta + f (\nabla \psi \cdot \nabla \theta) \nabla \phi \quad (7)$$

Note, however, that \hat{e}_θ does not lie completely in a toroidal cross section.

We also want to look at the perturbed magnetic field, $Q = \nabla \times (\xi \times B)$, which can be measured with Mirnov loops. We find

$$\begin{aligned}
Q = & \left[\frac{Jf}{X^2 g R} (\nabla \psi \cdot \nabla \theta) \frac{\partial g}{\partial \psi} \xi_\theta^P + \frac{Jg}{X^2 R} (\nabla \psi \cdot \nabla \theta) \frac{\partial}{\partial \psi} \left(\frac{f \xi_\theta^P}{g} \right) + \frac{Jf}{X^2 R} |\nabla \theta|^2 \frac{\partial \xi_\theta^P}{\partial \theta} \right. \\
& + \frac{f^2}{X^2 g R} \frac{\partial \xi_\theta^P}{\partial \phi} + \frac{J^2 B^2}{X^2 g R^2} \left(|\nabla \theta|^2 \frac{\partial \xi_\theta^P}{\partial \phi} - i \nabla \psi \cdot \nabla \theta \frac{\partial \xi_\theta^P}{\partial \theta} \right) \left. \right] \nabla \psi \\
& - \left[\frac{Jf}{X^2 g R} |\nabla \psi|^2 \frac{\partial g}{\partial \psi} \xi_\theta^P + \frac{Jg}{X^2 R} |\nabla \psi|^2 \frac{\partial}{\partial \psi} \left(\frac{f \xi_\theta^P}{g} \right) + \frac{Jf}{X^2 R} (\nabla \psi \cdot \nabla \theta) \frac{\partial \xi_\theta^P}{\partial \theta} \right. \\
& + \left. \frac{J^2 B^2}{X^2 g R^2} \left(\nabla \psi \cdot \nabla \theta \frac{\partial \xi_\theta^P}{\partial \phi} - i |\nabla \psi|^2 \frac{\partial \xi_\theta^P}{\partial \theta} \right) \right] \nabla \theta \\
& + \left[\frac{f \xi_\theta^P}{g R^2} \left\{ \frac{\partial}{\partial \psi} \left(\frac{Jf}{X^2} |\nabla \psi|^2 \right) + \frac{\partial}{\partial \theta} \left(\frac{Jf}{X^2} \nabla \psi \cdot \nabla \theta \right) \right\} + \frac{f}{R^2} |\nabla \psi|^2 \frac{\partial}{\partial \psi} \left(\frac{f \xi_\theta^P}{g} \right) \right. \\
& - \left. \frac{X^2}{g R^2} \left\{ \frac{\partial}{\partial \psi} \left(\frac{J B^2 \xi_\theta^P}{g} \right) + i \frac{\partial}{\partial \theta} \left(\frac{J B^2 \xi_\theta^P}{g} \right) \right\} + \frac{f^2}{g R^2} (\nabla \psi \cdot \nabla \theta) \frac{\partial \xi_\theta^P}{\partial \theta} \right] \nabla \phi
\end{aligned} \quad (8)$$

The most useful components of Q for experimental measurements are

$$Q_\psi = \frac{Q \cdot \nabla \psi}{|\nabla \psi|}, \quad Q_\theta = \frac{Q \cdot (\nabla \phi \times \nabla \psi)}{|\nabla \psi| |\nabla \psi|}, \quad (9)$$

where the component Q_θ now lies in a toroidal cross section. For an $n = 0$ case these components reduce to

$$Q_\psi = \frac{2\pi f}{v X^2 |\nabla \psi|} \frac{\partial \xi_\theta^P}{\partial \theta}, \quad (10)$$

$$Q_\theta = -\frac{\xi_\theta^P}{X R} \left[\frac{\partial \ln \xi_\theta^P}{\partial \psi} + \frac{\partial \ln f}{\partial \psi} \right] - \frac{f}{X R |\nabla \psi|} (\nabla \psi \cdot \nabla \theta) \frac{\partial \xi_\theta^P}{\partial \theta}, \quad (11)$$

which can be directly compared with experimentally measured values. For example, Q_θ vs θ on the outer flux surface is plotted in figure 2 for the instability case of figure 1. Q_ψ and Q_θ may also be Fourier decomposed to look at the mode structure. The Fourier components of Q_θ for an $m = 3$ surface kink mode are given in figure 3.

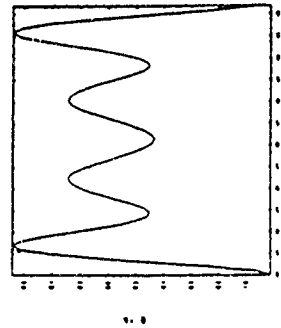


Figure 2

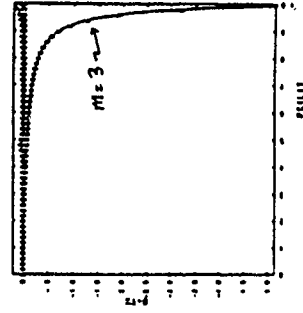


Figure 3

Measurements of the perturbed field at the Mirnov loops can be predicted from these components by noting that the m th component of the field decays as $r^{-(m+1)}$

$$Q_{\theta, mn}(r) = Q_{\theta, mn}(a) \left(\frac{a}{r} \right)^{m+1} \cos(m\theta + \pi\phi). \quad (12)$$

Contours of constant Q_ψ and Q_θ are also useful for illustrating mode structure. The $m = 3$ structure for this kink mode can be clearly seen in the contours of Q_ψ in figure 4

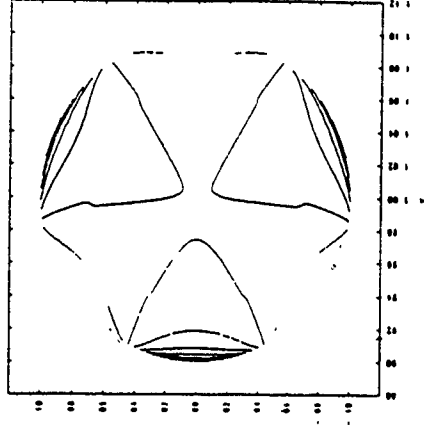


Figure 4 Contours of Q_ψ

The post-processor can also be used to determine how well the incompressibility assumption, $\nabla \cdot \xi = 0$, is satisfied. This is done by calculating

$$\begin{aligned}
\nabla \cdot \xi = & \frac{1}{X^2 R} \frac{\partial (X^2 \xi_\psi^P)}{\partial \psi} + \frac{i}{X^2 R} \frac{\partial (X^2 \xi_\theta^P)}{\partial \theta} + \frac{v f}{2\pi g R^3} \left[\nabla \psi \cdot \nabla \theta \frac{\partial \xi_\psi^P}{\partial \phi} - i |\nabla \psi|^2 \frac{\partial \xi_\theta^P}{\partial \phi} \right] \\
& + \frac{i 2\pi R f}{v X^2} \frac{\partial \xi_\psi^P}{\partial \theta} + \frac{i R g}{X^2} \frac{\partial \xi_\theta^P}{\partial \phi}
\end{aligned} \quad (13)$$

Thus for a given tokamak equilibrium and with the eigenvectors returned by the PEST code, we can now construct quantities with proper normalization which may be directly measured experimentally. Application of this program has already proven to be useful in analysis of data from PBX-M plasmas?

*This work supported by DoE contract No. DE-AC02-76-CFO-3073

¹Much of this work performed under appointment to the Magnetic Fusion Science Fellowship program, U.S. Department of Energy

²R. C. Grimm, J. M. Greene, and J. L. Johnson, Methods in Computational Physics **16**, 253 (1976)

³D. W. Roberts, Ph. D. thesis, Princeton University, 1991

GRAY SCALE RENDERING OF BEAM PROFILE DATA

H. Rudd, RCS, Lanham, MD 20706

I. Haber, Naval Research Laboratory, Washington, DC 20375

Particle-in-cell simulation has become a reliable tool for scientific research in many areas of Beam Physics. However, the usefulness of such computer simulation can be substantially enhanced by diagnostic methods which allow the numerical results to be directly compared to the experimental observations.

The present work resulted from the on-going numerical simulation support for the Electron Beam Transport Experiment at the University of Maryland [1, 2]. The transport system consists of 36 periodically spaced solenoidal magnets with two magnetic lenses used to match the beam. The transverse structure of the beam can be observed at any point in the channel by using a moveable phosphor screen.

The simulation studies have been performed using the two-dimensional particle-in-cell code SHFT-XY [3]. In the past, when the experiment was concerned with transporting a single 5 kV, 240 mA solid beam, scatter plots of the particle distribution in the X-Y plane seemed a sufficient visualization tool for representing the transverse beam structure in the simulation. However, when the experiment was changed to transport multiple beams, a complex, highly nonuniform beam structure was discovered which required a better method of rendering beam profile data.

In the multiple beam experiments, five small beamlets are masked out of the full 240 mA beam at the gun. During transport, the five beamlets merge into an almost uniform single beam at a distance of 14-17 cm. An interesting and unexpected effect was the appearance of images of the initial five-beamlet configuration between 98 and 112 cm downstream from the mask. Both merging and imaging of the beams in the experiment were followed through a series of phosphor screen pictures taken at short intervals and recorded with a charge-coupled video camera. The output was digitized on an Apple Macintosh computer, and the digital images were then displayed and photographed.

It was desired to render the simulated cross sections in a format which could be used for direct comparison with the fluorescent screen data. Scatter plots were found to be inadequate for reproducing the details of rapidly changing transverse beam configuration. Even with a number of simulated particles far in excess of what was necessary to get the correct physics, point plots were unable to reproduce the complexities of the dynamical beam structure because of the saturation which occurs as points are plotted on top of each other and therefore do not visually increase beam density. Small variations in beam structure that are evident in the experimental photographs can be very difficult to show in these simulation plots. On the other hand, contour plots of the same simulated beam profile data accentuate small fluctuations in the simulated beam density which do not appear on the actual photographs. Furthermore, false color fill of the adjacent contours increase this effect.

A successful solution to this problem came with the use of TeX Gray scale rendering of numerical beam profile data, binned into a 200X200 grid. A gray scale is attributed to each level, and bins are plotted in the appropriate gray shade. The initially granular visual reproduction of the beam cross sections can be further improved by a simple algorithm which smoothes the particle distribution over the 9 neighboring bins.

Fig. 1 shows corresponding experimental (a) and simulated beam profiles (b, c) recorded at 3.4 cm and 101.0 cm.

At 101 cm, the images of the initial beam configuration are clearly visible on the experimental photograph as well as on both simulation plots. However, the gray scale rendering of the data (Fig. 1, b) is obviously superior to the point plot (Fig. 1, c) in reproducing the finer details of the beam structure. By filling in the valleys between the beam intensities, gray scale visualization most closely resembles the way our human eye averages gray levels in perceiving a black and white picture. Reversing the gray scale (black = level 0) works to our advantage when reproducing the fluorescent screen diagnostics shown.

In all fairness, one disadvantage of the gray scale data rendering should be noted here: these pictures do not photocopy well. Reproduction of reduced xeroxed plots in this paper does not adequately display the degree of agreement between the experiment and simulation which has been observed in original plots.

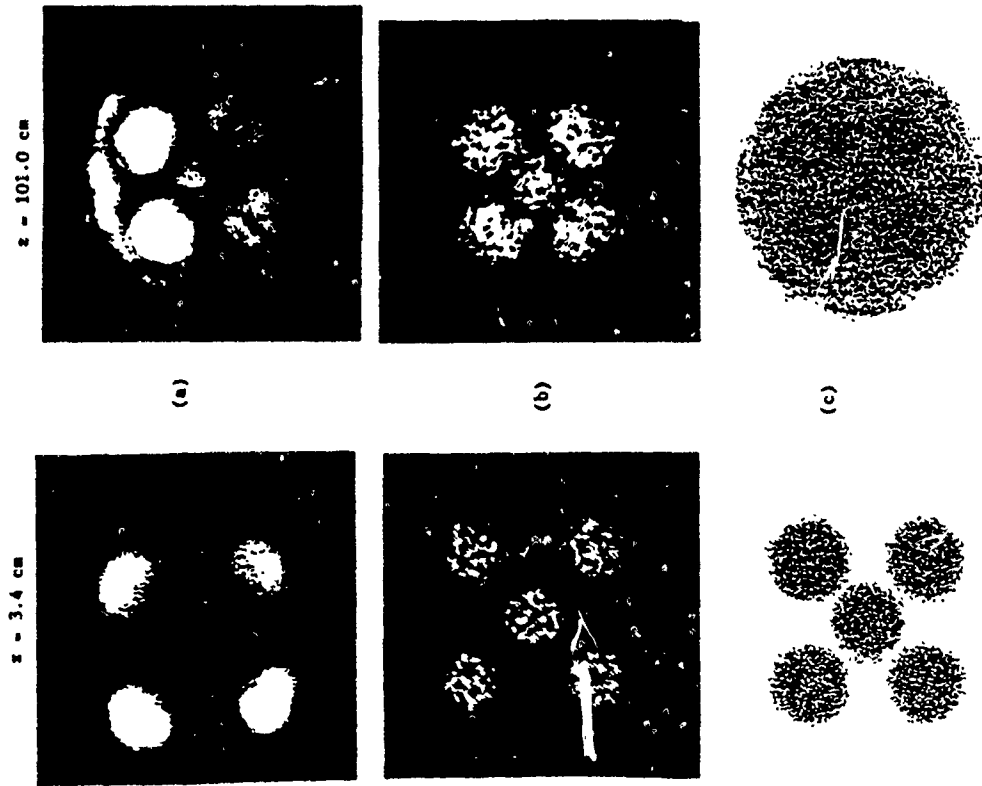


Fig. 1. Beam profiles at 3.4 cm and 101 cm downstream from the anode as they appear in (a) phosphor screen photographs; (b) gray scale rendering of simulated data, and (c) point plot with 32K particles shown.

The gray scale rendering of simulated beam profiles became an invaluable tool for detailed comparison between experiment and simulation in many subsequent studies [4, 5]. It enabled us to correctly adjust several parameters to reach an excellent agreement in the location, shape and orientation of the images. The computer renderings permitted us to reproduce the fine structure of the beam at any location in the transport channel.

By virtue of their surprisingly close resemblance to the experimental photographs gray scale renderings give further credibility to the numerical simulation, which continues to be used to investigate a wide parameter range for the Maryland Electron Beam Transport Experiment.

Acknowledgements.

We would like to thank Dr. Horace Mitchell for providing us with his implementation of TeX gray scale plotting technique for numerical data. This research was supported by the U.S. Department of Energy.

References

- [1] M. Reiser, C. R. Chang, D. Kehne, K. Lov, T. Shea, H. Rudd, and I. Haber, "Emittance Growth and Image Formation in a Nonuniform Space-Charge Dominated Electron Beam," *Physical Review Letters*, Vol. 61, No. 26, pp. 2933-2936, 1988.
- [2] I. Haber, D. Kehne, M. Reiser, and H. Rudd, "Experimental, Theoretical, and Numerical Investigation of the Homogenization of Density Nonuniformities in the Periodic Transport of a Space-Charge Dominated Beam," accepted for publication in *Physical Review A*.
- [3] I. Haber, H. Rudd, "Numerical Limits on PIC Simulations of Low Emittance Transport," in *Linear Accelerators and Beam Optics Codes - La Jolla Institute*, 1988, edited by C. R. Emlinizer, AIP Conf. Proc. 177, (A.I.P. New York, 1988) p. 161.
- [4] D. Kehne, M. Reiser, and H. Rudd, "Experimental Studies of Emittance Growth Due to Initial Mismatch of a Space Charge Dominated Beam in a Solenoidal Focusing Channel," to be published in the proceedings of the 1991 Particle Accelerator Conference, San Francisco, California.
- [5] D. Kehne, M. Reiser, and H. Rudd, "Experiments and Simulation of Emittance Growth due to Mismatch of a Space-Charge Dominated Electron Beam," to be published in the proceedings of the 1991 Symposium on High Brightness Beams for Advanced Accelerator Applications, College Park, Maryland.

XGRAFIX: AN X-WINDOWS ENVIRONMENT FOR REAL-TIME INTERACTIVE SIMULATIONS

Vahid Vahedi and John P. Verboncoeur
University of California, Berkeley
Berkeley, CA 94720

We have developed a real-time user interface environment, XGRAFIX, for simulations running under X-Windows. XGRAFIX is written in C in an object oriented style, and since it uses only the lower level X-Windows function-calls, it can be compiled with any superset of X-Windows, e.g. Motif, and is compatible with many systems. XGRAFIX allows the user to display multiple diagnostics and view them as they evolve in time. Like most other X environments, XGRAFIX provides keyboard and mouse support. The simulation codes are structured as shown in Fig. 1

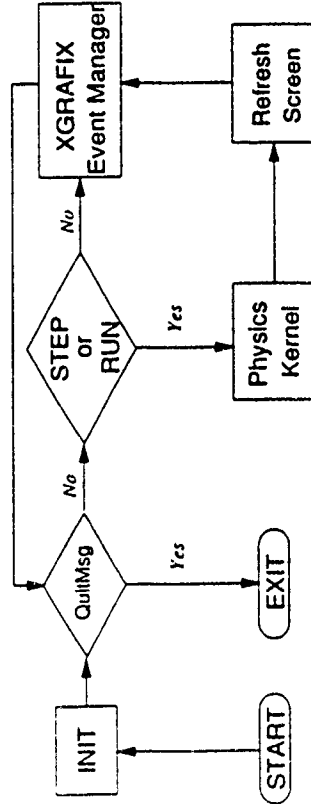


Figure 1 Schematic representation of the interaction between XGRAFIX and the physics kernel

The physics kernel is portable to any machine supporting standard C. The INIT module scans the input file containing the physical parameters of the problem and initializes the diagnostic windows. INIT also sets up memory for array storage. The environment provides hooks for the physics kernel to run continuously (there is no time limit - the code can run indefinitely) or step through individual timesteps. The windows are refreshed each timestep, and all user requests are processed by the XGRAFIX Event Manager. When the simulation is in the running state, the

Event Manager is invoked at each time step to process the events; if not running, the Event Manager is constantly invoked. The events include moving, resizing, and iconifying windows as well as mouse button clicks and keyboard inputs. XGRAFIX also supports PostScript output of the plots.

The environment currently offers three types of plots: linear, semi-log, and scattered. XGRAFIX provides the user with menus, dialogboxes, and smart windows. Each window has (currently) four standard buttons for rescaling the graph, viewing traces of plots, PostScript output, and a crosshair for measurements. The menus, dialogboxes, and the mouse make the environment especially user-friendly.

XGRAFIX is being currently rewritten in C++ which offers pre-defined classes and hierarchies used in the object-oriented style for such objects as buttons, menus, windows, etc. This modification should make it easier to handle future additions to XGRAFIX. We are also adding optimized three dimensional plotting routines to the environment which will make XGRAFIX even more useful for 2D and 3D simulations.

We are presently running three of our codes, ES1 (Electrostatic 1 Dimensional periodic plasma simulation)¹, PDP1 (Plasma Device, Planar 1 Dimensional)² and PDP2 (Plasma Device, Planar 2 Dimensional) in XGRAFIX.

REFERENCES

- ¹ C. K. Birdsall and A. B. Langdon, *Plasma Physics via Computer Simulation* (McGraw-Hill, New York, 1985).
- ² I. J. Morey, V. Vahedi and J. Verboncoeur, *Particle Simulation Code for Modeling Processing Plasmas*, *Bull. APS*, 34 2028 (Abstract 1989)

EQUIDISTRIBUTION STRATEGIES FOR PARTICLE CODES

by

Wayne Arter
Culham Laboratory, Abingdon, Oxon, OX14 3DB, England
(UKAEA/Euratom Fusion Association)

1 INTRODUCTION

Particle simulations may suffer from clumping, i.e. inadequate sampling of the field they represent, caused by a non-uniform spread of particle positions and/or momenta. In cases where particles are allowed to have e.g. different masses rather than all the same, we may try to alleviate clumping by equidistribution. This entails introducing a more uniform distribution of particles in position and/or velocity space, with new masses chosen to reflect the concentrations represented by the original set. Clearly there is a multitude of possible equidistribution strategies, we have to decide when to equidistribute, whether to do so locally in a few trouble spots or globally, throughout the whole computational domain, and further to determine which properties the strategy should preserve and to what accuracy.

We treat equidistribution in the context of an Extended Particle-in-Cell (EPIC) code designed to simulate the nonlinear interactions of drift waves. Since drift-wave turbulence is often spatially homogeneous, we concentrate on global equidistribution strategies. Many of the lessons learnt should apply to most particle codes, however.

2 MODEL PROBLEM

The prototype for many drift-wave studies is the Hasegawa-Mima Equation, also known as the "equivalent Barotropic Equation in geophysical applications,

$$\left(\frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla\right) F = 0, \quad (1)$$

where $\mathbf{u} = \nabla \Phi \wedge \hat{\mathbf{z}}$, and $F = \Phi - \nabla^2 \Phi + \delta_n y = \rho + \delta_n y$

The velocity potential Φ is a function of x, y and t only, $\hat{\mathbf{z}}$ is the unit vector in the z direction and δ_n is the dimensionless density gradient. More realistic models of drift-waves involve other effects that require the n th particle to have attribute F_n as well as position \mathbf{x} .

In our EPIC scheme [1] particle positions obey $\dot{\mathbf{x}} = \mathbf{u}(\mathbf{x}, t)$. To calculate gradients of Φ , EPIC invokes a dual representation for $\rho (= F - \delta_n y)$, viz

$$\rho(\mathbf{x}) = \left\{ \sum_p \rho_p \delta(\mathbf{x} - \mathbf{x}_p), \right. \quad (2)$$

1

where $\delta(\mathbf{x})$ is the Dirac delta function and $\psi_p(\mathbf{x})$ is a basis function corresponding to node p of some discretisation. The two representations (2) are made equivalent by taking their inner product with $\psi_q(\mathbf{x})$, so

$$\sum_p \rho_p (\psi_p, \psi_q) = \sum_p \rho_p \psi_q(\mathbf{x}_p). \quad (3)$$

Eq(3) can be written schematically as

$$MM\rho = \hat{\rho} = AM(\rho), \quad (4)$$

where MM denotes the mass matrix (ψ_p, ψ_q) , $\hat{\rho}$ is lumped ρ , and AM represents the so-called charge assignment operator. Our equidistribution strategy leaves unchanged ρ_p , that may be straightforwardly calculated from the ρ .

3 THE ONE-DIMENSIONAL CASE

For simplicity, take $\psi_p(x)$ to be the chapman function, viz for nodes p of unit spacing, $\psi_p(x) = \psi(x+p)$ where

$$\psi(x) = \begin{cases} 1+x, & -1 \leq x \leq 0, \\ 1-x, & 0 < x \leq 1, \end{cases} \quad (5)$$

MM then has the stencil $(\frac{1}{4}, \frac{3}{4})$ and AM represents area weighting. Let ρ_p^* denote the masses of particles at x_p^* ; we seek to replace these by particles of mass ρ_p at specified, uniformly distributed positions x_p , using $\rho_p^* = AM(\rho)$.

The obvious approach is to work out the contributions of the particles within an element to ρ_p at each of its nodes. Provided the number of particles per node $J \geq 2$, we can choose ρ_p so that $\rho_p^* = AM(\rho)$. The disadvantage is that extra storage of size $2^d N$ is needed, where N is the number of nodes and d is the dimension.

No extra storage is needed when we construct the operator assigning mass to particles AP ($\rho_p = AP(\rho_p^*)$) such that $AM \cdot AP = MM$. AP is defined in terms of weighting factors $\alpha_1, \dots, \alpha_J$, such that particle j between nodes C and L is assigned a mass $\rho_p = \alpha_j \rho_C + \alpha_{j+1} \rho_{C+1}$. In Fig 1(a), where nodes are midway between particles (case MM'), the total contribution of the ρ_j to C is

$$\frac{(\rho_L + \rho_H)}{J} \sum_{j=1}^J \alpha_j (J-j+\frac{1}{2}) + \frac{\rho_C}{J} \left\{ \sum_{j=1}^J \alpha_j (J-j+\frac{1}{2}) + \alpha_{J-j+1} (J-\frac{1}{2}) \right\}. \quad (6)$$

Equating (6) to $MM\rho$ shows that any $\{\alpha_i\}$ satisfying

$$\sum_{j=1}^J \alpha_j = \frac{1}{2}, \quad \sum_{j=1}^J j \alpha_j = \frac{J}{6} + \frac{1}{4}. \quad (7)$$

will be acceptable. Eq (7) has no solutions when $J = 1$; for $J = 2$, the factors are unique, viz $\{\alpha_1, \alpha_2\} = \{\frac{1}{4}, \frac{1}{4}\}$; for $J = 3$ and higher there are infinite possibilities

2

When $J = 3$ the choice $\{\alpha_i\} = \{\frac{1}{2}, \frac{1}{3}, \frac{1}{6}\}$ is preferred, since then adjacent equal p_i generate equal p_j . We note the importance of MM : if it is lumped, i.e. MM is replaced by the Identity I , then for $J = 2$ we find $3\alpha_2 + \alpha_1 = 0$, so one weighting factor must be negative. This seems very undesirable.

Tuning to the case AN where there is a particle at each node: let the nodal particle have mass $\alpha_0 p_0 + \alpha_1(p_L + p_R)$. Assign factors $\alpha_1, \dots, \alpha_{J-1}$ to the other particles as in the MW case, then a similar analysis yields

$$\sum_{j=1}^J \alpha_j = \frac{1}{2}(1 - \alpha_0), \quad \sum_{j=1}^J \alpha_j = \frac{J}{6}. \quad (8)$$

When $J = 1$, $\{\alpha_0, \alpha_1\} = \{\frac{1}{3}, \frac{1}{3}\}$ is the obvious solution; for $J = 2$ the equal p_i constraint makes $\{\alpha_i\} = \{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\}$.

4 TWO DIMENSIONS

The one dimensional results are of great utility in higher dimensions because as the operators AM^{-1} and MM factor into products of 1-D operators, so does AP . Figure 1(b) shows how we can use the product rule $\alpha_i = \alpha_{i,1} \alpha_{i,2}$ to determine weighting factors in the 2-D, MW case. $\{\alpha_{i,j}\}$ for AN may be determined similarly, and it is straightforward to calculate $\{\alpha_{i,j}\}$ for a distribution of x_i , which is e.g. AN in z and MW in y .

5 TEST CASE

$Eq(1)$ is solved in a 125×83 domain, with $\Delta x_i = 1$, for random initial conditions with an energy spectrum $\propto k^{-2}$. The numerical parameters employed were $J = 4$, $N = 384$ and $\Delta t = 1$. Initially energy E , measured as $\frac{1}{2} \sum p_i^2 \phi_i$, was 11125. Without equidistribution, particles are seen to bunch together, resulting in very messy ϕ contours and an E that falls steadily to 5468 after 500 timesteps.

Equidistribution strategies based on the maximum and minimum number of particles (n_{max} and n_{min} respectively) allowed in a cell were tried. Particle bunching is eliminated if $n_{max} = 2J$; for that run, the final E was 10636 and 10 equidistributions were performed using MW particles.

6 REFERENCES

- [1] W Arter and J W Eastwood, "Particle Mesh Schemes for Advection Dominated Flows", submitted to J. Comp. Phys. (1991)

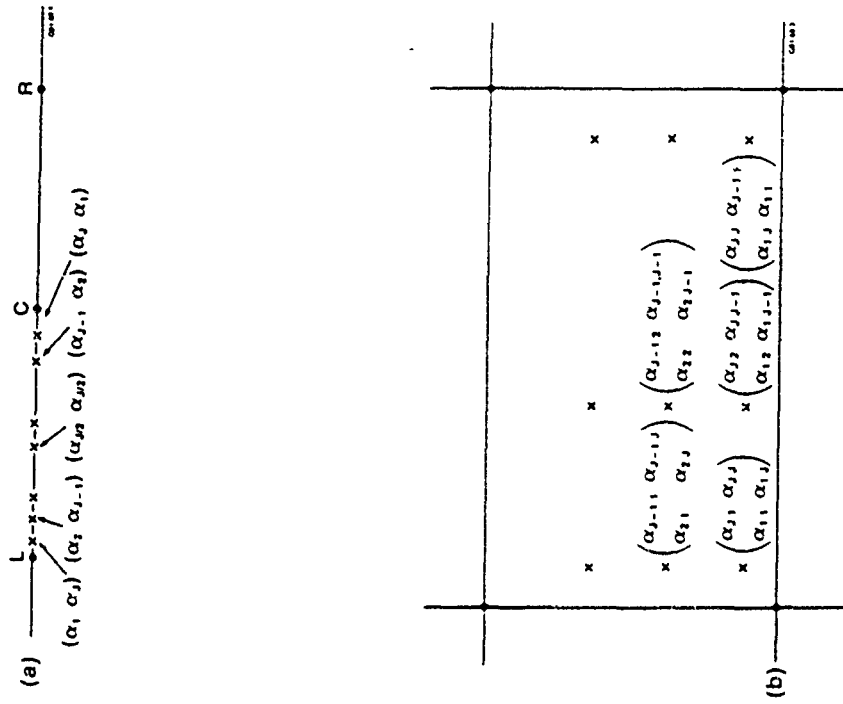


Figure 1. Assignment of mesh-based information to particles. Particle positions are marked with crosses, dots indicate nodes. (a) One-dimensional case, (b) two-dimensional case.

Quiet Implicit PIC

D. C. Barnes and R. A. Nebel
Los Alamos National Laboratory, F-647
Los Alamos, NM 87545

We describe a new algorithm for the implicit electromagnetic simulation of nearly collisionless plasmas. In this Quiet Implicit Particle-In-Cell (QIP) method, the unknown distribution function f is separated into a bulk portion f_0 and an intrinsically kinetic portion δf . Single-simulation-particle shot noise is greatly reduced by treating only the intrinsically kinetic portion using PIC methods. The bulk portion, on the other hand, is determined by the solution of moment equations coupled to Maxwell's equations and differenced time-implicitly. The QIP method is fully kinetic, nonlinear, electromagnetic, and appropriate for domains of arbitrary dimensionality with general boundary conditions. The QIP method is a generalization of previous " δf " developments.^{1, 2} Generalizations reported here include local variation of the bulk distribution in both space and time, a fully electromagnetic formulation, and implicit time differencing.

The separation of f from δf leads to an enormous noise reduction, in the case when δf is small compared to f . For example, an intrinsic noise level of less than 10^{-4} has been observed using 10 particles per cell. Such quiet, long-time-step simulations allow the investigation of previously inaccessible kinetic phenomena, such as fluctuation-driven particle and energy transport in magnetically-confined plasmas, where observed fluctuation levels are a few 10^{-3} , and fluctuation time scales are 10^5 times longer than particle orbit time scales.

To derive the simulation equations, the distribution f_α for each species α is written

$$f_\alpha = f_{\alpha M} + \delta f_\alpha, \quad (1)$$

where $f_{\alpha M}$ is a local Maxwellian whose low velocity moments agree with those of f_α .

¹ M. Kotschenreuther, et al., Paper No. D-4-16, 13th International Conference on Plasma Physics and Controlled Nuclear Fusion Research, Washington, DC, 1-6 October, 1990.

² A. Dimits and B. I. Cohen, Personal Communication, 1991.

Explicitly,

$$f_{\alpha M} = \frac{\rho_\alpha}{(2\pi T_\alpha/m_\alpha)^{3/2}} \exp\left(-\frac{m_\alpha(v_\alpha - u_\alpha)^2}{2T_\alpha}\right), \quad (2)$$

where the number density ρ_α , fluid velocity u_α , and temperature T_α are separate functions of position r and time t for each species, and where m_α is the species mass.

As indicated, the moments of $f_{\alpha M}$ are chosen to agree with those of f_α . Thus, these moments satisfy the usual low-order moment equations

$$\frac{\partial \rho_\alpha}{\partial t} + \nabla \cdot \rho_\alpha u_\alpha = 0, \quad (3a)$$

$$\frac{\partial(\rho_\alpha u_\alpha)}{\partial t} + \nabla \cdot \rho_\alpha u_\alpha u_\alpha = \frac{Q_\alpha \rho_\alpha}{m_\alpha} E + \rho_\alpha u_\alpha \times \Omega_\alpha - \nabla k_B \rho_\alpha T_\alpha - \nabla \cdot \Pi_\alpha, \quad (3b)$$

$$\frac{\partial(\rho_\alpha T_\alpha)}{\partial t} + \nabla \cdot \rho_\alpha T_\alpha u_\alpha = -(\gamma - 1) \left(\rho_\alpha T_\alpha \nabla \cdot u_\alpha + \frac{\nabla u_\alpha \cdot \Pi_\alpha + \nabla \cdot Q_\alpha}{k_B} \right), \quad (3c)$$

where E is the electric field, $\Omega_\alpha = \frac{Q_\alpha B}{m_\alpha c}$ the cyclotron frequency vector, Π_α the traceless part of the stress tensor, and Q_α the heat flux, with Q_α the species charge and k_B Boltzmann's constant.

Equations (3) are, of course, not closed, since Π_α and Q_α are not determined. On the other hand, these quantities are independent of $f_{\alpha M}$, and depend only on δf_α . Our approach is to advance a collection of simulation particles in the usual PIC manner to obtain δf_α at the phase-space location of these particles. Then the desired moments are computed by interpolating this information to the mesh used to advance the moment equations. The particle equations of motion are the usual leap-frog differenced Newton-Lorentz equations. To determine δf_α at the particle's phase-space location, one additional quantity is carried by each particle. This is the value of the initial ($t = 0$) distribution function on the particle's orbit. Since the Vlasov equation requires that the total f be constant along any orbit, δf_α may then be computed directly. Actually, evaluation of the required moments of δf_α involve sums of $\frac{\delta f}{f}$, assuming that particles are loaded in the conventional way so that a sum over particles represents a phase-space integral over f . Thus, knowing the phase-space coordinates of particle j ,

$$\left(\frac{\delta f_\alpha}{f_\alpha}\right)_j = 1 - \left(\frac{f_{\alpha M}}{f_\alpha}\right)_j, \quad (4)$$

Notice that all quantities on the right of (4) may be evaluated directly. In fact, f_0 is a constant carried by the j th particle, and f_{OM} may be evaluated by interpolating the moment information from the mesh to the j th particle using its position and velocity.

The formulas for evaluation of Π_0 and q_0 are

$$\Pi_{0j} = \frac{\nu}{\Delta V_g} \sum_i s(\mathbf{x}_j - \mathbf{x}_i) \left(\frac{\delta f_0}{f_0} \right) (w_j w_j - w_j^2/3), \quad (5a)$$

where the subscript g refers to the grid point \mathbf{x}_g , where s is the interpolation function from the particle position \mathbf{x}_j to the grid, where ΔV_g is the volume element associated with \mathbf{x}_g , ν a constant depending on the number of physical particles per simulation particle, I the unit dyadic, and where $w_j = v_j - u_j$ is the single particle peculiar velocity. Similarly

$$q_{0j} = \frac{\nu}{\Delta V_g} \sum_i s(\mathbf{x}_j - \mathbf{x}_i) \left(\frac{\delta f_0}{f_0} \right) w_j (m_0 w_j^2/2 - 5/2 k_B T_0). \quad (5b)$$

The summands of (5) have been adjusted to account for certain systematic errors which occur in the calculation of δf . A correction has been made on the basis that the zeroth, first, and trace of the second velocity moments of δf are required to vanish. Because of truncation errors, these constraints will not be satisfied exactly. Thus we make the Ansatz that the numerically determined δf differs from the exact by the local Maxwellian multiplied by a linear combination of 1 , w_j , and w_j^2 . Using the constraints, the coefficients of this linear combination are determined and give the form (5), where δf there is the value determined by (4).

Equations (3), coupled with Maxwell's equations are advanced implicitly in time, as in the implicit moment PIC method.³ Since (3) depend on those quantities derived from the particle data in (6), it is necessary to use a predictor-corrector approach. Thus, using the old time step E , the particles are advanced to evaluate (5). Then a predictor advance of (3) is done. Next, the error in the implicit Maxwell equations is evaluated, and E corrected to the current time step. Finally, moments are corrected to be consistent with (3) and particle positions and velocities are corrected to account for the change in E .

³ J. U. Brackbill and D. W. Forslund, in *Multiple Time Scales*, J. U. Brackbill and D.

Cohen, Eds. (Academic Press, Orlando, 1985), pp. 271-310.

We have confirmed several rather dramatic advantages associated with employing the moment equations in our formulation. First, it is straightforward to arrange the conservation of charge, momentum, and energy, since these properties depend only on the centering of quantities on the mesh, and consistency between the various members of (3). This permits us to drop Gauss's law and $\nabla \cdot \mathbf{B} = 0$ from Maxwell's equations and avoid the need to perform divergence cleaning or other charge correcting machinations. Second, linear stability depends only on (3) combined with Maxwell's equations. Thus, there is great flexibility in evaluating the particle quantities (5). For example, we have confirmed that substepping the particles and averaging (5) over these substeps in a first-order accurate manner does not adversely affect stability. Finally, use of the moment equations somewhat simplifies the treatment of complex geometries and boundary conditions.

We have implemented QIP in one and three-dimensions. In the one-dimensional case, a nonlinear instability associated with errors in evaluation of the heat flux has been observed. This instability is associated with the loss of synchronization between the representation of low-order moments by the particle list and by the moment list. Thus, for example, truncation errors in the time advance may lead to a case in which no particles are present in one portion of the mesh, while the moment list indicates a significant density there. These instabilities may be avoided by enforcing synchronization between the two representations. We have found that an effective strategy is to modify the particle list to conform to the moment list. This produces a smooth solution and avoids the nonlinear instability mentioned previously. The optimum method of synchronization is still being investigated, and results will be reported.

We will also report one-dimensional tests of the method. In one series of such tests, a compressional Alfvén wave is launched at one boundary and propagates through a uniform, magnetized plasma toward the other, reflecting boundary. When the magnetic field has a component in the nonignorable direction, the resulting k_z induces electron Landau damping of the wave. We will present a quantitative comparison of the observed damping with theoretical expectations. We will also report initial three-dimensional calculations carried out in a periodic, circular cylinder geometry. In these calculations, the internal $m = 1$ kink of a current-carrying plasma column, will be examined, and kinetic modifications to the fluid expectations observed.

Modeling the Longitudinal Wall Impedance Instability

in Heavy Ion Beams Using an R-Z PIC Code*

D. A. Callahan, A. B. Lington, A. Friedman, and D. P. Grote, Lawrence Livermore National Laboratory, Livermore CA 94550, and I. Haber, U.S. Naval Research Laboratory, Washington DC 20375

Abstract

The effects of the longitudinal wall impedance instability in a heavy ion beam are of great interest for heavy ion fusion drivers. This instability occurs when a beam travels down a pipe with finite impedance. In a heavy ion fusion driver, this impedance is due to the accelerating modules of the induction LINAC. The effects of this instability are important because final focusing places requirements on the beam quality. We have developed a model for the resistive and capacitive parts of the impedance and have implemented it in the R-Z thread of the WARP PIC code. We will use this code to explore the effects of finite capacitance and finite length beams on this instability.

Model of the Resistive Wall

First we consider the purely resistive case. In the long wavelength limit ($\lambda \gg 2\pi b$ where b is the wall radius), linear theory predicts a forward traveling wave ($\omega_{real} = +kv_p$) and a backward traveling wave ($\omega_{real} = -kv_p$). The forward traveling wave damps with decay rate $\omega_i = -\eta v_b v_p / 2g$ while the backward traveling wave grows with growth rate $\omega_i = \eta v_b v_p / 2g$ where η is the resistance per length, v_b is the beam velocity, and g is a geometric factor.

We wish to develop a model for the resistive wall in the code. We would like to formulate a scheme which includes the resistive wall in the calculations without explicitly using the beam current. We want to avoid estimating the current through cross sections, i.e. a highly localized current. Involving the Poisson solve lets us, in effect, estimate a current but as some weighted average over a volume of pipe of length \sim beam radius, and we hope that will be more physical and smoother.

We begin with a continuity equation for the wall surface charge, σ , which has units of charge/area

$$\frac{\partial \sigma}{\partial t} + \frac{\partial K_z}{\partial z} = 0 \quad (1)$$

where K_z is the surface current. We also have Ohm's law, $2\pi b \eta K_z = E_z$, where η is the resistance per unit length. Plugging this into the continuity equation yields

$$\frac{\partial \sigma}{\partial t} = -\frac{\partial}{\partial z} \left(\frac{E_z}{2\pi b \eta} \right) \quad (2)$$

Integrating this equation over the surface and assuming cylindrical symmetry gives

$$\frac{\partial}{\partial t} \int_{z_1}^{z_2} \sigma dz = -\frac{1}{2\pi b \eta} [E_z(z_2) - E_z(z_1)] \quad (3)$$

where we have taken the limits on z to be z_1 to z_2 . Define the electrostatic potential, ϕ , by

$$E_z(z') = -\frac{\partial \phi}{\partial z} \Big|_{z'} \quad (4)$$

Consider the case of $z_2 = z_1 + \Delta z$ where Δz is small. Over this range, we can assume that σ doesn't change much and we can evaluate the integral over σ giving us

$$\frac{\partial}{\partial t} \int_{z_1}^{z_1 + \Delta z} \sigma dz = \Delta z \frac{\partial \sigma}{\partial t} \quad (5)$$

Defining the surface charge, sQ , as ${}^sQ = 2\pi b \Delta z \sigma$, and using the results of equations (4) and (5) in equation (3) gives

$$\frac{\partial ({}^sQ)}{\partial t} = \frac{1}{\eta} \left[\frac{\partial \phi}{\partial z} \Big|_{z_1 + \Delta z} - \frac{\partial \phi}{\partial z} \Big|_{z_1} \right] \quad (6)$$

Since we have assumed that Δz is small, we can expand $\partial \phi / \partial z$ in a Taylor series. Taking the first two terms of this expansion and substituting into equation (6) gives the result

$$\frac{\partial ({}^sQ)}{\partial t} = \frac{\Delta z}{\eta} \frac{\partial^2 \phi}{\partial z^2} \quad (7)$$

This equation is then solved simultaneously with Poisson's equation in the code.

The solve is done by Fourier transforming equation (7) with respect to z and then using a finite difference approximation to the time derivative. Doing this gives,

$${}^s\tilde{Q}^{n+1} = {}^s\tilde{Q}^n - \frac{\Delta t \Delta z}{\eta} k^2 \tilde{\phi}^{n+1} \quad (8)$$

where the superscript denotes the time level and the tilde denotes the Fourier transform. We can write Poisson's equation as

$$(\nabla_r^2 - k^2) \tilde{\phi}^{n+1} = -\frac{\tilde{\rho}^{n+1}}{\epsilon_0} \quad (9)$$

where ∇_r^2 is the radial part of ∇^2 . At the wall, $\tilde{\rho}^{n+1}$ is composed of the charge density due to the plasma plus the charge density due to the surface charge, ${}^s\tilde{Q}^{n+1}$. We substitute for ${}^s\tilde{Q}^{n+1}$ in Poisson's equation at the wall from equation (8) and

are left with an equation for ϕ^{n+1} in terms of the surface charge at the last time step, Q^n , and the plasma density. The radial part of the Laplacian is differenced to give a tridiagonal system of equations for each value of k . These systems are solved via Gaussian elimination using the boundary condition that radial electric field $E_r = -\nabla\phi/\nabla r = 0$ outside the wall. This Neumann boundary condition gives the solution for the potential to within an arbitrary constant. Since the quantities of interest are the electric fields, we can arbitrarily set the constant to zero at the wall. We then update the surface charge using equation (8) to complete the field solve.

Model Including Capacitance

For a resistor and capacitor in parallel, we get Ohm's law

$$2\pi b\eta K_s = E_s + \eta C \frac{dE_s}{dt} \quad (10)$$

where b is the pipe radius, η is the resistance per unit length, and ηC is the "RC" time. We can then go through the same derivation with this Ohm's law that we did when we had resistance alone. This gives the result:

$$\frac{\partial \sigma}{\partial t} = \frac{1}{2\pi b} \frac{\partial}{\partial z} \left[\frac{1}{\eta} \left(\frac{\partial \phi}{\partial z} + \eta C \frac{d}{dt} \frac{\partial \phi}{\partial z} \right) \right] \quad (11)$$

where σ is the surface charge density and has units of charge/area. By saving ϕ at the previous time step, we can use a finite difference approximation for the time derivatives and then solve simultaneously with the Poisson equation. We also define the surface charge, $Q = 2\pi b \Delta z \sigma$

In the case of η and ηC constant, we can Fourier Transform this equation in z and we get

$$\frac{Q_k^{n+1} - Q_k^n}{\Delta t} = -\frac{\Delta z}{\eta} k_z^2 \left(\phi_{r,z,k}^{n+1} + \eta C \frac{\phi_{r,z,k}^{n+1} - \phi_{r,z,k}^n}{\Delta t} \right) \quad (12)$$

where Q and ϕ are the Fourier transforms of Q and ϕ .

Theories done in one dimension have shown that even a small amount of capacitance significantly reduces the growth of the instability².

¹A. Friedmann, D. A. Callahan, D. P. Grote, A. B. Langton, and I. Haber, Proc. of the Conf. on Computer Codes and the Linear Accelerator Community, Los Alamos, NM, January 21-25, 1990, R. K. Cooper and K. C. D. Chan, eds.

²E. P. Lee and L. Smith, Proc. of the Linear Accelerator Conference, Albuquerque, NM, September 10-14, 1990.

*Work performed in part under the auspices of the U.S. Dept. of Energy by Lawrence Livermore National Laboratory under contract W-7405-ENG-48.

finite-difference operator into two 1-D ones in X and Y directions. In order to find successive approximations in each direction a version of the multi-sweep algorithm with FFT procedure is used to overcome nonlinearity in boundary conditions.

Numerical results. The space-time discharge structure was calculated for following parameters:

$$n_N(x, y, 0) = 10^{10} \text{ cm}^{-3}; T(x, y, 0) \approx 300 \text{ K}; \lambda = 0.8 \text{ cm}; \theta = 60^\circ;$$

The obtained numerical results show that the discharge in the electromagnetic wave beams field rapidly arrives to the nonuniform stage. The field and plasma parameters distribution at the moment $t_* = 2.5 \text{ nks}$ are given in Figs. 2, 3. The maximum value of temperature T by this time is 340 K. The great gas heating rate ($\max_{x,y} dT/dt|_{t=t_*} = 30 \text{ K/nks}$) is caused by the quenching of the excited

electron states of molecules. From the calculated electrical field values, the absorption factor is obtained as a function of time. Note that electromagnetic waves energy is absorbed mainly into the strongly nonuniform plasma structures, where it is most effectively transformed into the internal energy of the gas. A good qualitative agreement of numerical results with experimental data [3] was observed.

References:

1. Silakov V.P. Sov. Plasma Phys., 1988, v.14 M0, p.1209-1213.
2. Gear C.W. Comm. Assoc. Comput. Mach. 1971, v.14, p.185-190.
3. Viharev A.L. et al. Sov. Plasma Phys., 1984, v.10, M., p.165-168.

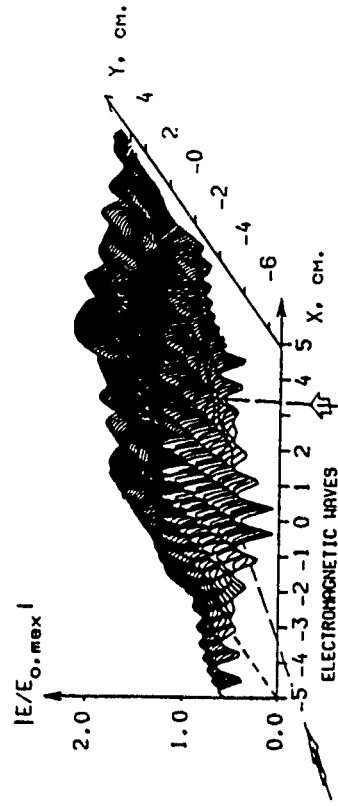


Fig 2.

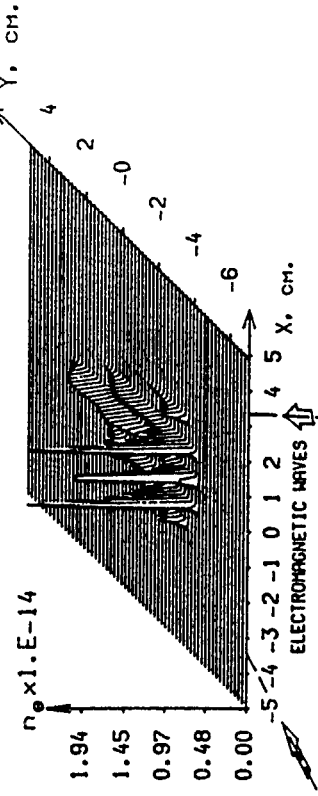


Fig 3.

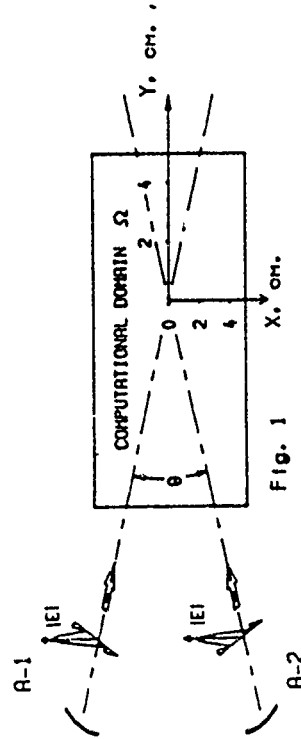


Fig. 1

The Stability of Non-Coplanar Rotational Discontinuities

P. J. Cargill and C. C. Goodrich

Department of Physics and Astronomy, University of Maryland, College Park, MD 20742

Rotational discontinuities (RDs) are one of the four fundamental classes of discontinuities that can exist in a plasma (Hudson, 1970), the others being shocks, contact discontinuities (CDs) and tangential discontinuities (TDs). It is a relatively simple exercise to show that shocks and CDs satisfy a so-called coplanarity theorem which means that the magnetic field on either side of the discontinuity must lie in the same plane. However, RDs differ from both shocks and CDs in that a rotation of the magnetic field out of the coplanarity plane is permitted as one crosses the discontinuity. An example of this is an RD that rotates a magnetic field such that (in a cartesian system) the field is $\mathbf{B} = (B_x, 0, B_z)$ on the upstream side and $\mathbf{B} = (B_x, B_y, 0)$ on the downstream side. Here we assume that the RD varies in the normal direction (x coordinate) only, so that $B_x = \text{const}$. This rotation can be accomplished in two ways. Either the field can rotate through 90° with B_y and B_z both > 0 , or it can rotate through 270° with B_y and B_z both becoming negative at some point. The rotation through 90° is accomplished by a wave-like structure that has a sense of rotation corresponding to the Alfvén (intermediate) mode and the 270° rotation corresponds to the fast mode wave. The question of whether both of these rotations are stable is central to our understanding of the structure of RDs.

We have performed 1-D hybrid simulations of such non-coplanar RDs. The present hybrid code treats the ions as simulation particles and the electrons as a massless, infinitely conducting fluid with a polytropic index of $5/3$. When the magnetic field rotates out of the coplanarity plane by means of a 90° rotation, the RD relaxes in a manner similar to the coplanar case by shedding fast (Alfvén) waves in the upstream (downstream) direction (Goodrich and Cargill, 1991). It would thus appear that RDs with a 90° rotation are stable, irrespective of whether the initial sense of rotation corresponds to the fast or Alfvén modes.

The results are rather more interesting for the 270° rotations. When \mathbf{B} rotates from the positive z direction upstream to the positive y direction downstream (fast mode sense of rotation), the initial 270° RD relaxes to one that rotates the field through 90° in the Alfvén sense. This is accomplished by the shedding of a large amplitude fast mode wave pulse that propagates away slowly in the upstream direction. This pulse involves a 360° rotation of the magnetic field, so that the total helicity of the system is conserved. The pulse is elliptically polarized, and maintains its form over many ion cyclotron periods, suggesting that its nature is similar to that of the well known solitary waves. The width of this wave is typically 20 ion inertial lengths.

When RDs that involve a transition between a positive B_z and a negative B_y , the 270° solution corresponds to the Alfvén sense of rotation. Hybrid simulations have again shown that this is unstable, with evolution toward a 90° fast mode sense of rotation. The evolution away from the initial state is much more rapid than in the previous case. There is no evidence of an upstream wave pulse, but the helicity-conserving evolution requires the shedding of a 360°

Alfvén pulse which would be unable to propagate upstream in any event. There is some evidence of a large amplitude Alfvén pulse on the downstream side, but it proves to be relatively hard to distinguish it from the other waves located there.

This work has applications to both the solar wind and the Earth's magnetopause. RDs are a common feature of the solar wind, but the experimental data shows little evidence of RDs in which the magnetic field rotates by in excess of 180° . The same result holds at the magnetopause. The present work suggests that if RDs are formed with rotations in excess of 180° , a rapid relaxation should take place to a more stable state. One sign of such a relaxation would be the appearance of large amplitude pulses with a 360° rotation of the magnetic field.

C. C. Goodrich and P. J. Cargill, *Geophys. Res. Lett.*, **19**, 65, 1991.

Numerical Simulation of Electron Dynamics in Inductive Multigap Accelerators and Light Ion Beam Focusing*

B. W. Church, D. W. Longcope, B. V. Oliver, and R. N. Sudan

Laboratory of Plasma Studies, Cornell University

We investigate two problems in exploring the potential of intense light ion beams as a driver for inertial confinement fusion (ICF). Extremely high power, in excess of 10^{13} watts for 20-30 nanoseconds is required to generate such beams powerful enough to initiate fusion in deuterium-tritium pellets. The beams are produced in ion diodes^{1,2} which are energized by pulsed power sources. The first part of this work focuses on the complexities of magnetic insulation in multigap inductive accelerators which are being investigated for use as these pulsed power sources. The HERMES III accelerator³ at Sandia National Laboratory is the largest of this class of accelerators, producing a 20 megavolt, 700 kilamp pulse for 30 nanoseconds. We study the magnetic insulation of electron flow in these accelerators in order to determine the working principles such that an optimally efficient design of the HERMES-type machine can be achieved. Because the electrons emitted in different gaps have different energy and canonical momenta, the simple one dimensional, single component theory⁴ must be extended to such multi-component flows. To gain an understanding of these flows, we simulate multi-component electron flow in Hermes-type devices using a finite element code FERUS for dimensional particle in cell code MASK⁵. In addition, we are developing a finite element code FERUS for finding quasi-static electron flows in nontrivial geometries. The second problem we study has application in an achromatic transport and focusing system⁶, proposed by Oloso⁶, for the purpose of focusing a light ion beam from an initial width of $\sim 10 - 15$ cm on to a pellet less than 1 cm in diameter. An important component of the system is a short, plasma filled, applied B solenoidal lens. The plasma fill provides charge and current neutralization of the beam, thus allowing for ballistic transport and focusing. We present an extension of the initial calculations by including the effects of self-fields which can arise because the plasma electron and energetic beam ion dynamics are very dissimilar in the presence of the strong lens fields. We are developing a hybrid PIC code to solve the model equations.

In order to understand the working principles for the HERMES-type machine the PIC code MASK has been used to study the effects of load impedance on the distribution of electron flows and on the coupling power to the load in an accelerator with a small number of gaps for various load impedances. These results are compared with a simple theoretical model for multi-component electron flows. In order to understand quasi-static equilibria for these flows, a mathematical model based on fluid equations has been developed for the two dimensional flow of these electrons in the quasi-static limit. The novel finite element code FERUS is being developed to solve the equations of this model. The model is described below for flow in a coaxial transmission line.

Let the magnetic field, $\mathbf{B} = B(x, y, z)\hat{z}$ be oriented along the direction of the ignorable coordinate $(\partial/\partial y = 0)$. For a coaxial transmission line $y \rightarrow \theta$ the azimuthal coordinate. The cartesian model represents a TEM wave propagating in the x direction of a coaxial transmission line such that the annular width $d \ll r_0$ the radius of the inner conductor. We are justified in assuming that the transit time of the electron, $d/v = t_1 \ll \tau$ the rise time of the electromagnetic pulse. This enables the neglect of the $\partial/\partial t$ term in the momentum balance equation for the electron which may now be written in the quasi-static limit (in cgs units) as

$$(\nabla \cdot \nabla)\gamma v = \frac{e}{m}(\nabla\phi - \frac{1}{c}v \times B) \quad (1)$$

with $\gamma = (1 - v^2/c^2)^{-1/2}$ and the electric field $\mathbf{E} = -\nabla\phi(x, z)$ in the quasi-static limit. This equation is supplemented with Ampere's law

$$\frac{c}{4\pi} \nabla \times \mathbf{B} = \frac{e}{4\pi} \nabla B \times \hat{y} = -nev \quad (2)$$

where the electron current density is $-nev$, and with Poisson's equation

$$\nabla^2\phi = 4\pi ne \quad (3)$$

where $-ne$ is the charge density. Define $\omega = g \nabla \times \gamma v/c$ the vorticity of the electron flow. (From now on factors of c will be used to express frequencies in units of inverse length.) Then equation (1) may be rewritten as

$$\nabla(\phi - \gamma) + \frac{1}{c}v \times g(\omega - \Omega) = 0 \quad (4)$$

with $\Omega = eB/mc^2$ and $\phi = e\phi/mc^2$. From (4) and (2) we obtain the relations

$$v \cdot \nabla \Omega = 0 = v \cdot \nabla(\phi - \gamma) \quad (5)$$

i.e. Ω and $(\phi - \gamma)$ are constant along streamlines of electron flow. Let all the electrons be emitted from the cathode surface, $\phi = 0$, with zero velocity. This implies that for all electrons $\phi = \gamma - 1$, which defines energy conservation and consequently from (4) we conclude that $\omega - \Omega = 0$ on each streamline. From equation (2) it is straightforward to establish that

$$\omega^2 = \frac{4\pi e^2 n}{mc^2} = \frac{\gamma}{\sqrt{\gamma^2 - 1}} |\nabla \Omega| \quad (6)$$

From equations (2), (3) and (6) it follows that

$$\nabla^2 \gamma = \omega^2 = \frac{\gamma}{\sqrt{\gamma^2 - 1}} |\nabla \Omega| \quad (7)$$

$$\nabla \left(\frac{\gamma}{\omega^2} \nabla \Omega \right) = \nabla \left(\sqrt{\gamma^2 - 1} \frac{\nabla \Omega}{|\nabla \Omega|} \right) = \omega - \Omega \quad (8)$$

Equations (7) and (8) constitute a self-consistent set that describes the quasi-static electron flow in the cold fluid approximation. In the one-dimensional limit (obtained by setting $\partial/\partial z = 0$) we can recover both the Child-Langmuir solution and relativistic Brillouin flow⁷. These equations are non-linear PDEs of non-standard type; to study these a non-standard numerical method has been developed based on a finite element expression of the equations. The domain can be decomposed into an electron flow region, \mathcal{E} , and a vacuum region, \mathcal{V} . Equation (7) applies to the entire domain although its right hand side vanishes in \mathcal{V} ; equation (8) applies only to region \mathcal{E} . Since the boundary curve, $\partial\mathcal{E}$, is an unknown along with the fields $\Omega(x)$ and $\gamma(x)$ it is convenient to use a numerical technique with a movable mesh.

In the finite element implementation of the equations $\Omega(x)$ and $\gamma(x)$ are prescribed at each node, x_i , of a triangular mesh and defined by linear interpolation inside each triangle. A set of basis functions, $\{\eta_i(x)\}$, also linear within each triangle, are defined by the relation $\eta_i(x_i) = \delta_{ij}$, forming irregular "tent" over the node x_i . The space of piece-wise linear functions from which Ω and γ are drawn is spanned by these basis functions

$$\Omega(x) = \sum_i \Omega_i \eta_i(x) \quad \gamma(x) = \sum_i \gamma_i \eta_i(x) \quad (9)$$

As the mesh is refined so that the use of each individual triangle vanishes, the space of piece-wise linear functions will approach the space of continuous functions, and the numerical equations will approach the analytic ones.

Cast equation (8) in the weak form by multiplying it by N_r basis functions in turn and integrating each product over \mathcal{E} . The N_r resulting equations are

$$\int_{\mathcal{E}} dx \left\{ \sqrt{\gamma^2 - 1} \frac{\nabla \Omega}{|\nabla \Omega|} \cdot \nabla \eta_r + \Omega \eta_r \right\} - \oint_{\partial\mathcal{E}} dl \sqrt{\gamma^2 - 1} \frac{\nabla \Omega}{|\nabla \Omega|} \cdot \partial \eta_r = 0 \quad (10)$$

Choosing N_r basis functions which all vanish on the boundary of the domain equation (7) can similarly be cast in the weak form:

$$\int_{\mathcal{E}} dx \left\{ \nabla \gamma \cdot \nabla \eta_r + \frac{\gamma}{\sqrt{\gamma^2 - 1}} |\nabla \Omega| \eta_r \right\} = 0 \quad (11)$$

As unknowns choose a single component from each of N_r node positions and the value γ at N_r interior nodes; the value of Ω will remain fixed on every node. The movable nodes must be from region \mathcal{L} thus allowing the entire region to move as the equations are solved.

The program PERUS uses an $N_r + N_r$ dimensional Newton's method to solve equations (10) and (11) from a sufficiently good initial guess. All of the nodes in region \mathcal{L} except those on the cathode are allowed to move and the potential is fixed to $\gamma = 1$ on the cathode and $\gamma = V + 1$ on the anode. Since the value of Ω fixed on each node the total current carried by the electrons is a prescribed quantity. Runs have been done with up to 200 unknowns exploring two types of solutions: vertical uninsulated Child-Langmuir-like flow, and horizontal self-insulated Brillouin-like flows.

Our second area of investigation concentrates on the transport of light ion beams generated by a HERMES III type device. Analysis has identified a solenoidal magnetic lens system as a possible scheme⁸. The system assumes a charge and current neutral beam by filling the transport channel with plasma. However, because the plasma fill will have a finite conductivity, currents may arise which will distort the applied lens fields. In order to address this problem we have developed a self-consistent model of magnetic field evolution during beam passage.

Beam ions are considered to be single particles. Their orbits are governed by the Lorentz force

$$M_i \frac{d\mathbf{v}}{dt} = Ze(\mathbf{E} + \frac{\mathbf{v} \times \mathbf{B}}{c}) \quad (14)$$

We model the plasma as being composed of heavy immobile ions and collisional fluid electrons. The condition of no background ion motion requires that the pulse duration time τ be short enough that $v_A \tau \ll r_0$, where v_A is the Alfvén speed and r_0 is the beam radius. We are not concerned with high frequency (Langmuir) oscillations as electron inertia is neglected. Quasi-neutrality $n_e = Z n_0 + Z_i n_i$ is also presumed.

The plasma electrons are described by the generalized Ohm's law

$$\mathbf{j}_e = \sigma(\mathbf{E} + \frac{\mathbf{v}_e \times \mathbf{B}}{c}) \quad (15)$$

where \mathbf{j}_e is the electron current, σ the plasma conductivity, and \mathbf{v}_e the electron fluid velocity. By taking the curl of (15) and utilizing Faraday's law

$$\nabla \times \mathbf{B} = -\frac{1}{c} \frac{\partial \mathbf{B}}{\partial t} \quad (16)$$

an induction equation for \mathbf{B} is given by

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times \mathbf{v}_e \times \mathbf{B} - \nabla \times \eta(\nabla \times \mathbf{B} - \frac{4\pi}{c} \mathbf{j}_e) \quad (17)$$

where the Darwin limit of Ampere's law

$$\mathbf{j}_e = \frac{c}{4\pi} \nabla \times \mathbf{B} - \mathbf{j}_b \quad (18)$$

has been substituted on the r.h.s of Eq (16). \mathbf{j}_b is the beam current, and $\eta = \frac{c^2}{4\pi\sigma}$ is the magnetic diffusivity. We consider the two-dimensional axisymmetric ($\frac{\partial}{\partial \theta} = 0$) case, and represent the magnetic field in Grad-Shafranov coordinates by writing

$$\mathbf{B} = \nabla \psi \times \nabla \ell + B_\theta r \nabla \theta, \quad (19)$$

where $\psi = r A_\theta$ and $\nabla \ell = \frac{1}{r} \nabla \phi$. By substitution of Eq (17), we obtain

$$\frac{\partial \psi}{\partial t} + v_r \nabla \psi = \eta \Delta_\perp \psi + \frac{c^2}{4\pi} \frac{\partial \psi}{\partial r}, \quad (20)$$

from the θ component of Eq (13), and from the ℓ component of Eq. (15).

$$\frac{\partial B_\theta}{\partial t} + (v_r \nabla) B_\theta = \frac{\eta}{r} \Delta_\perp r B_\theta + r(B_\theta \cdot \nabla)(\frac{v_\theta}{r}) - (\nabla \cdot \frac{v_\theta}{r}) r B_\theta + \frac{c}{4\pi} \nabla \cdot \nabla \psi \quad (21)$$

where the operator $\Delta_\perp = r^2 \nabla \cdot \nabla$

Eqs (18) and (19) comprise a set of coupled nonlinear PDE's, which can be used to advance the magnetic field in time. The set is complete by virtue of the fact that given the beam ion source terms, the time evolution of ψ and B_θ are functions of only themselves. The electron velocity \mathbf{v}_e is derived from Eq (16) where $\mathbf{j}_e = -en_e \mathbf{v}_e$. The corresponding components are given by

$$v_{e\theta} = \frac{1}{n_e c} \left(-\frac{c}{4\pi} \Delta_\perp \psi + j_b \right), \quad (22)$$

and

$$v_{e\ell} = \frac{1}{n_e c} \left(j_b \ell - \frac{c}{4\pi} \nabla(r B_\theta) \times \nabla \ell \right) \quad (23)$$

The electric field \mathbf{E} , necessary for the particle advancement in Eq. (12), is obtained from Eq. (13). The four Eqs. (18, 19, 20, 21) together with the beam ion momentum Eq.(12) are solved self-consistently with a hybrid, PIC code. Our objective is to understand the lens focusing properties as a function of beam parameters and the plasma conductivity. This system of equations is advantageous since we are free from having to solve the quasi-neutral Poisson equation. In addition, compared to codes which solve for the electric field \mathbf{E} from (13) and advance the three components of \mathbf{B} using Faraday's law, we need only advance the two scalar variables ψ and B_θ , all other field quantities are derivable from them. Standard numerical diffencing has been used throughout and the time advance of Eqs. (18) and (19) is facilitated by an explicit low order Euler method, making for a simple robust code. The simulated physics is similar to the CHDER⁹ code.

- 1 S. Hunphries and R. N. Sudan, US Patent No. 405677
- 2 S. Hunphries, Nucl. Fusion, 20, 1849 (1980)
- 3 D. L. Johnson et al., "HERMES-III Positive Polarity Experiment", *Proceedings of the 7th IEEE Pulsed Power Conference*, 1989.
- 4 C. W. Jr. Mendel, D. B. Seidel, D. B. Rosenthal, Laser and Particle Beams, vol. 1, part 3, p. 311 (1981)
- 5 A. T. Drobot, Bull. Am. Phys. Soc. 29, 1379 (1984)
- 6 C. L. Olson, in *Proc 1988 Linear Accelerator Conference*, CEBAF, Oct. 3-7 1988
- 7 R. N. Sudan, Bull. Am. Phys. Soc. Paper 7RV, 34, 2111 (1989)
- 8 J. M. Creedon, J. App. Phys., 46, 2946 (1975); R. V. Lovelace, E. Ott, Phys. Fluids 17, 1263 (1974)
- 9 A. Mankofsky, R. N. Sudan, J. Donavit, J. Comp. Phys. 70, 89 (1987)

*Research supported by Sandia National Laboratory, Albuquerque, under Contract No. 63-4881 and by the Plasma Physics Division of the Naval Research Laboratory under ONR Contract N00014-85-J-1770

COMPARISON OF 3-D VMEC AND PIES CODE STELLARATOR EQUILIBRIUM CALCULATIONS

J. L. Johnson, D. A. Monticello, and A. H. Reiman
Plasma Physics Laboratory, Princeton University*

A. Salas
Asociación EURATOM-CIEMAT para Fusión, Madrid

S. P. Hirshman
Oak Ridge National Laboratory†

The PILS code¹ is a magnetohydrodynamic code that determines three-dimensional toroidal equilibrium configurations by direct integration along the field lines. It can thus study systems with magnetic islands and multiple magnetic axes. This code has, until now, been used mainly for study of problems associated with tokamak configurations: investigations of the nonlinear saturation of tearing modes, the behavior of magnetic surfaces in the presence of error fields, etc. The VMEC code,² on the other hand, determines the equilibrium properties of such configurations by minimizing the energy. It is based on an 'inverse solver' technique in which a set of well behaved nested magnetic surfaces must be postulated to exist. It has been the principal tool for the determination of equilibrium configurations for the design of stellarators.

Comparisons of the calculations of the equilibrium properties of stellarators by these two different approaches would provide additional validation for both codes and thus make additional tools available for the study of complex configurations. We report the first results of such a study for two devices: the ATF toratron at Oak Ridge and the TJ-II heliac which is being built in Madrid. Detailed results from calculations for vacuum and low- β equilibria with good magnetic surfaces in both systems show excellent agreement.

In this work we specify the shape of an outer boundary of the plasma by giving the number of field periods N , the major radius of the system R_{Major} , and the parameters $R_{m,n}$ and $Z_{m,n}$, such that

$$R(\theta, \phi) = R_{Major} + \sum R_{m,n} \cos(m\theta - nN\phi), \quad (1)$$

$$Z(\theta, \phi) = \sum Z_{m,n} \sin(m\theta - nN\phi), \quad (2)$$

and choose our pressure distribution to be

$$p(s) = p_0(1 - s)^2; \quad (3)$$

$$s \equiv \frac{\psi - \psi_{axis}}{\psi_{edge} - \psi_{axis}}$$

where ψ is the poloidal flux inside the surface.

ATF with $< \beta > = 0$

The ATF toratron at Oak Ridge has a major radius of 2.1 m and a minor average radius of 27 cm. The 2 T magnetic field is produced by helical windings with a poloidal multiplicity of $\ell = 2$ and $M = 12$ toroidal field periods. Auxiliary poloidal field coils are available to modify the shape of the configuration, but we treated the standard one where the vacuum field rotational transform varies between $\epsilon \approx 0.3$ at the magnetic axis and $\epsilon \approx 1.0$ at the plasma boundary. We carried through some comparisons of this equilibrium using several different mesh sizes for each code.

Comparisons of the rotational transform, $q(s)$, and of the average magnetic well, given by $\{dV(s)/ds\}/\{d\psi(s)/ds\}$ where $V(s)$ and $\psi(s)$ are the volume and toroidal flux inside the surface s , are shown in Fig. 1 and Fig. 2. Here, we used $k = 45$ surfaces with $m = 0, \dots, 10$ poloidal modes and $n = -3, \dots, 3$ toroidal modes with the PIES code and $k = 45$ surfaces with $m = 0, \dots, 5$ poloidal modes and $n = -6, \dots, 6$ toroidal modes with the VMEC code. The extrapolation to infinite mesh of the rotational transform at the magnetic axis is shown in Fig. 3. This extrapolation indicates that larger numbers of radial mesh points must be used with the VMEC code than with PIES to get into the asymptotic domain where things should scale as $1/k^2$.

ATF with $< \beta > = 1\%$

In this comparison, we used the same plasma boundary as in the vacuum field case. The toroidal field at the plasma edge with $\theta = 0$, $\phi = 0$ was set equal to 2.265 T to make $\psi_{edge} = 0.25$ webers and the value of $p_0 = 0.0172$ newtons/m² was chosen so that $< \beta > \approx 1\%$.

The comparisons of the rotational transform, $q(s)$, and the magnetic well parameter $V'(s)$, are given in Figs. 4 and 5. Here, the PIES code had $k = 45$, $m = 0, \dots, 10$, and $n = -3, \dots, 3$. The VMEC code had $k = 60$, $m = 0, \dots, 5$, $n = -6, \dots, 6$. The extrapolation to infinite mesh of the rotational transform near the magnetic axis is shown in Fig. 6. The agreement is not perfect, with some of the problem being that large meshes are needed with the VMEC code to get into the asymptotic regime. There is also some dependence on the number of poloidal modes that are retained.

TJ-II

The TJ-II heliotron in Madrid is currently under construction. It will have a major radius of 1.5 m. The shape of the poloidal cross section varies with the toroidal position with the average plasma radius varying between 10 and 25 cm. The magnetic field of about 1 T is produced primarily by a set of toroidal field coils. The field is shaped largely by current in an axisymmetric poloidal coil at the center around which the toroidal field coils are staggered with an $M = 4$ toroidal periodicity. There are thus large $\ell = 0, 1$, and 2 magnetic field components, with higher- ℓ and higher- M toroidal moments also important. An additional helical coil around the central ring provides flexibility. We

investigate the 'shallow well' configuration in which s is nearly constant, rising from about $s \approx 1.46$ at the axis to $s \approx 1.48$ at the plasma edge.

The computations of the vacuum-field configuration by the two codes agree very well. Comparisons of the rotational transform, $\epsilon(s)$, and of the average magnetic well, given by $[dV(s)/ds]/[d\psi(s)/ds]$, where $V(s)$ and $\psi(s)$ are the volume and toroidal flux inside the surface s , are given in Fig. 7 and Fig. 8. We used $k = 60$, $m = 0, \dots, 5$, and $n = -12, \dots, 12$ with the PIES code and $k = 60$, $m = 0, \dots, 7$, and $n = -12, \dots, 12$ with the VMEC code for these figures. There is minor disagreement in the value of the rotational transform near the magnetic axis, but the problems are understood and are being corrected. Extrapolation of the magnetic axis position to infinite mesh size shows agreement to five significant figures.

Work is progressing on extending this comparison to TJ-II equilibria with finite β . Solutions have been found with $\beta > 0.1\%$ where the PIES and VMEC codes show good agreement.

REFERENCES

- * Supported by the U.S. Department of Energy under Contract DE-AC02-76-CHO-3073 with Princeton University.
- † Supported by the U.S. Department of Energy under Contract DE-AC05-84OR21400 with Oak Ridge National Laboratory, operated by Martin Marietta Energy Systems, Inc.
- ‡ A. Ilicman and H. Greenside, *J. Comput. Phys.* **78**, 1423 (1988).
- § S. P. Hirshman, W. I. van Rij, and P. Merkel, *Comput. Phys. Commun.* **43**, 143 (1986).
- || J. F. Lyons, G. L. Bell, J. D. Bell *et al.*, *Fusion Technology* **17**, 33 (1990).
- ¶ C. Alejandre, J. J. A. Gozalo, J. B. Perez *et al.*, *Fusion Technology* **17**, 131 (1990).

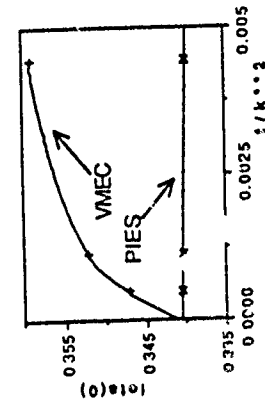


Fig. 3 $q(0)$ for the vacuum field ATF configuration as a function of mesh size

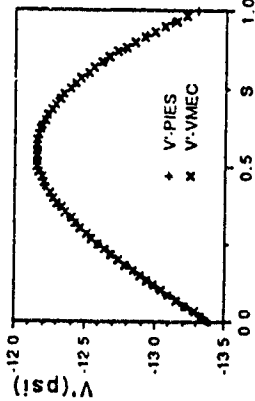


Fig. 5 $V'(\psi)$ vs. s for the $\langle \beta \rangle = 1\%$ ATF configuration.

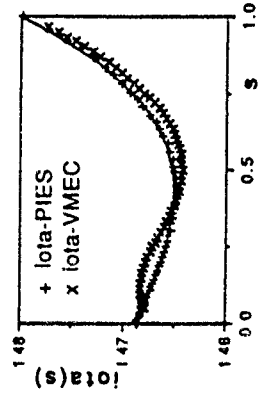


Fig. 7 $\epsilon(s)$ vs. s for the $\langle \beta \rangle = 0\%$ TJ-II configuration.

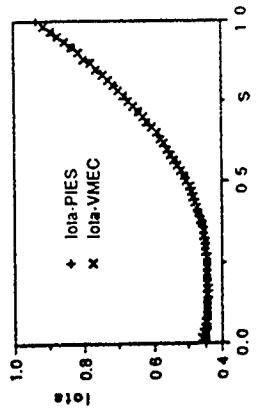


Fig. 4 $\epsilon(s)$ vs. s for the $\langle \beta \rangle = 1\%$ ATF configuration.

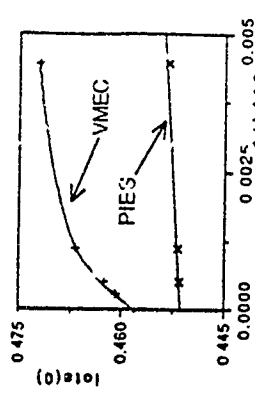


Fig. 6 $\epsilon(0)$ for the $\langle \beta \rangle = 1\%$ ATF configuration as a function of mesh size

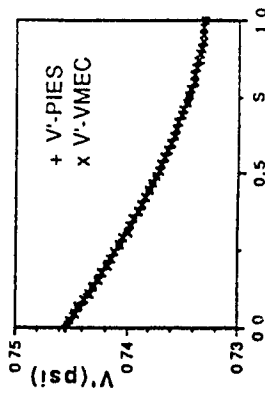


Fig. 8 $V'(\psi)$ vs. s for the $\langle \beta \rangle = 0\%$ TJ-II configuration.

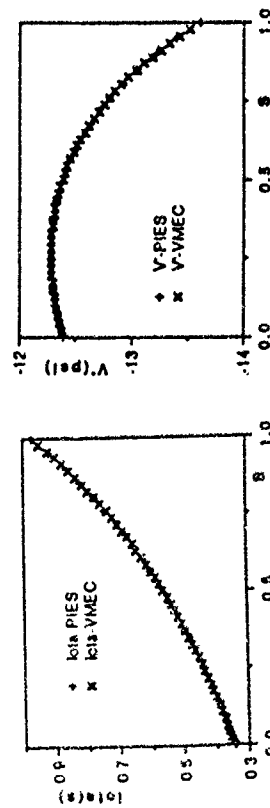


Fig. 1 $\epsilon(s)$ vs. s for the vacuum field ATF configuration

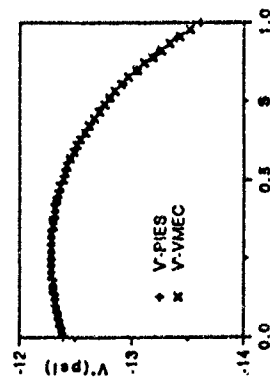


Fig. 2 $V'(\psi)$ vs. s for the vacuum field ATF configuration.

Semi-Implicit Compressible MHD Simulation with Gravity and Temporally Second-Order Convection

W. S. Lawson and H. R. Strauss

*Courant Institute of Mathematical Sciences
New York University*

*251 Mercer Street
New York, New York 10012*

Introduction

We are developing a code for use in MHD problems on the solar surface. Several methods have been developed to deal with the problems of separation of time scales, and the convective terms. The convective terms in particular are often treated poorly or ignored entirely, more, it seems, because they are difficult to deal with than because they are necessarily unimportant. We have attempted to solve this particular problem in developing a code that treats the convective terms accurately to second order. Some other methods have been used to increase speed which may be of interest.

General features of the code are: finite differenced in all directions; grid points are offset so as to minimize the amount of averaging, with its attendant diffusion; periodic boundary conditions allowed, but not necessary; semi-implicit time advancement; variable time step; dynamical variables are density, velocity, and vector potential (in the radiation gauge); physical effects include pressure, gravity, viscosity, and resistivity. Variable grid spacing is designed into the code, but has not been implemented.

Although the code development is not complete, the principles described here have been shown to work.

Time Advancement Scheme

Since a second-order convection scheme was desired, most standard techniques for the convective term were ruled out — even upwind schemes that are of high order spatially are only of first order in time. The scheme we have used couples the leap-frog scheme with a three-stage second-order Runge-Kutta scheme. While only two stages are necessary for second-order accuracy, three are needed for stability.

The leap-frog scheme is the usual one, with velocities known at half-integer time steps, and the density and vector potential known at integer time steps. The Runge-Kutta scheme is the following:

$$f^* = f^p - \frac{1}{2} \Delta t \mathbf{v} \cdot \nabla f^p, \quad (1a)$$

$$f^{**} = f^p - \frac{1}{2} \Delta t \mathbf{v} \cdot \nabla f^*, \quad (1b)$$

$$f^{p+1} = f^p - \Delta t \mathbf{v} \cdot \nabla f^{**}, \quad (1c)$$

where Δt is the time step, \mathbf{v} is the velocity, and f is any of the dynamical variables (including \mathbf{v} , for which superscripts of p , $*$, and $**$ must be added to the \mathbf{v} 's in the respective equations). This scheme is stable in the important special case of a constant velocity field, whereas the usual two-step scheme (which leaves out the middle step) is unstable. Several other schemes reduce to the same final result in this special case. The scheme (1) was chosen purely for ease of implementation, and its good performance in the non-linear case is not guaranteed. Other schemes may perform better, but this scheme has proved to be adequate.

Semi-Implicit Scheme

Rather than invert a two- or three-dimensional semi-implicit operator, as is usual, an operator was chosen that factors into one-dimensional

operators. This dubious-looking scheme is the following:

$$\psi^{p+1/2} = \psi^{p-1/2} + \Delta t \left[1 - C^2 \frac{\Delta t^2}{\Delta x^2} \Delta_{0x}^2 \right]^{-1} \left[1 - C^2 \frac{\Delta t^2}{\Delta y^2} \Delta_{0y}^2 \right]^{-1} F^p / \rho. \quad (2)$$

(The Runge-Kutta scheme is applied to this, but is omitted here for clarity) The linear dispersion relation for, say, sound waves using (2) is

$$\sin^2 \frac{\omega \Delta t}{2} = \frac{v_s^2 \frac{\Delta t^2}{\Delta x^2} \sin^2 \frac{k_x \Delta x}{2} + v_s^2 \frac{\Delta t^2}{\Delta y^2} \sin^2 \frac{k_y \Delta y}{2}}{\left[1 + v_s^2 \frac{\Delta t^2}{\Delta x^2} \sin^2 \frac{k_x \Delta x}{2} \right] \left[1 + v_s^2 \frac{\Delta t^2}{\Delta y^2} \sin^2 \frac{k_y \Delta y}{2} \right]}, \quad (3)$$

where the semi-implicit parameter C in (2) has been chosen optimally as $v_s/2$ (v_s is the sound speed). Clearly the restoring force in some of the high- k modes is too strongly attenuated, which would result in large excursions, and the problem will only become worse in three dimensions. A small viscosity, however, (which most investigators have found to be an empirical necessity anyway) will safely damp these modes, since they are all at high wave number. The savings in speed and simplicity are high.

Stabilization of Explicit Viscosity

A method brought to our attention by Harned (though unpublished, to the best of our knowledge) is used to stabilize the viscous terms (as well as the resistive and diffusive terms if desired). The method circumvents the usual (expensive) implicit solution. The principle of the method is the same as the principle of the semi-implicit method, namely, applying an operator to attenuate the Fourier components that will cause instability, while changing the long-wavelength components little. In our case, again to circumvent a two- or three-dimensional matrix inversion, the viscosity equation is:

$$\psi^{p+1/2} = \psi^{p-1/2} + \left[1 - 2 \frac{\mu \Delta t}{\Delta x^2} \Delta_{0x}^2 \right]^{-1} \frac{\mu \Delta t}{\Delta x^2} \Delta_{0x}^2 \psi + \left[1 - 2 \frac{\mu \Delta t}{\Delta y^2} \Delta_{0y}^2 \right]^{-1} \frac{\mu \Delta t}{\Delta y^2} \Delta_{0y}^2 \psi, \quad (4)$$

where μ is the coefficient of viscosity. Again, the accuracy of the high- k modes is sacrificed for speed and simplicity.

Boundary Conditions

Since the problem of interest (solar flares) is made much more approachable by allowing flow out through a boundary, sophisticated boundary conditions have been applied. Since the boundary conditions that allow structures to convect out of the system are unstable when the flow is into the system, different boundary conditions are used depending on the direction of the normal flow. Crudely speaking, when the flow is outward, guard cell values are linearly extrapolated and when the flow is inward, guard cell values are equal to the neighboring interior points. Unfortunately, there are many exceptions to this, and the means of extrapolating derived quantities, such as current density, have given a great deal of trouble. Although stable boundary conditions have been found, it would be dishonest to claim understanding at this point.

One set of troublesome boundary conditions are somewhat understood, and those are the conditions on the density in a gravitational field. Simple constant extrapolation at the upper boundary in the case of inward flow results in the excitation of sound waves. Since there is a natural scale height determined by the force of gravity and the compressibility, the density decay due to that scale height must be reflected in the extrapolation of the density.

LOW n MODE STABILITY ANALYSIS FOR $\ell=2$ HELIOTRON/TORSATRON BY VMEC-STEP CODE

Y Nakamura, K Ichiguchi and M. Wakatani

Plasma Physics Laboratory, Kyoto University, Uji, Kyoto, Japan

J L Johnson and G Rewoldt

Plasma Physics Laboratory, Princeton University, Princeton, New Jersey, USA

S. P. Hirshman

Oak Ridge National Laboratory, Oak Ridge, Tennessee, USA

Recent developments of three dimensional MHD equilibrium codes, such as the VMEC code, make full 3-D MHD equilibrium studies feasible. These exploit the periodicity associated with helical field periods in a heliotron/torsatron to reduce computation. The VMEC code is an inverse solver based on the energy minimization which uses the Fourier expansion in the poloidal and toroidal directions and the finite difference method in the radial direction.

Since 3-D stability studies with respect to modes that do not possess periodicity require enormous computation, they are time consuming. On the other hand, two-dimensional averaged codes should be efficient for surveying these stability properties of heliotron/torsatron. This is because the physical assumptions, namely the elimination of the fast magneto-sonic modes and the decoupling of modes with different toroidal mode numbers, that are made in the process of averaging over the field periods in the development of such models, apply well in heliotron/torsatron configurations. This has led us to develop techniques¹ to manipulate the output from the 3-D VMEC code² into a form where it could be used as input for the 2-D STEP stability code³.

In the VMEC code, the magnetic field is described by $\mathbf{B} \equiv \nabla s \times \nabla(\psi'\theta - \chi'\zeta - \lambda)$, where θ and ζ are poloidal and toroidal angle, ψ and χ are the toroidal and poloidal flux divided by 2π , respectively, primes denote the derivative with respect to a surface label s , and λ is a periodic function of θ and ζ . The toroidal angle ζ is chosen to be the geometrical toroidal angle ϕ in the VMEC code. Since this coordinate system (s, θ, ζ) is suitable for an equilibrium calculation but not useful for a stability analysis, mapping to a new coordinate system (s, θ, ζ) which has straight magnetic field line on the (θ, ζ) plane is done by a variable transformation $\theta = \vartheta - \lambda(s, \vartheta, \zeta)/\psi'$ and numerical root finding. On this new coordinate system, we apply the averaging procedure over the ζ direction, and obtain $(\bar{R}(s, \theta), \bar{Z}(s, \theta))$ and the effective magnetic field line curvature term $\Omega = (\bar{R}/R_0)^2 - (\bar{Z}/R_0 B_0)^2 - 1$, where

$B_s = B - \bar{B}$ and the bar means an averaged quantity. Inverse mapping of $(\bar{R}(s, \theta), \bar{Z}(s, \theta))$, that is $s(\bar{R}, \bar{Z})$, constructs averaged flux surfaces analogous to the lowest order flux surfaces in the stellarator expansion.³ Then the necessary metric quantities for analysis with the STEP stability code are

$$\begin{aligned} J &= R \left(\frac{\partial R \partial Z}{\partial \theta \partial s} - \frac{\partial R \partial \theta}{\partial s \partial \theta} \right), \\ |\nabla s|^2 &= \left(\frac{R}{J} \right)^2 \left(\left(\frac{\partial R}{\partial \theta} \right)^2 + \left(\frac{\partial Z}{\partial \theta} \right)^2 \right), \\ \nabla s \cdot \nabla \theta &= - \left(\frac{R}{J} \right)^2 \left(\frac{\partial R \partial R}{\partial s \partial \theta} + \frac{\partial Z \partial Z}{\partial s \partial \theta} \right), \\ |\nabla \theta|^2 &= \left(\frac{R}{J} \right)^2 \left(\left(\frac{\partial R}{\partial s} \right)^2 + \left(\frac{\partial Z}{\partial s} \right)^2 \right). \end{aligned}$$

In this averaged equilibrium configuration, the effective magnetic field is written as

$$\begin{aligned} \mathbf{B}_{eff} &= f(s) \nabla \phi + \nabla \phi \times \nabla s \\ &= s^{-1} \nabla s \times \nabla \theta + \nabla \zeta \times \nabla s. \end{aligned}$$

Here $f(s)$ and ζ should be consistent with the rotational transform s obtained by the VMEC code. If we define a phase difference between ζ and the geometrical toroidal angle ϕ by $\zeta = \phi - s^{-1} \delta(s, \theta)$, they are

$$\begin{aligned} \delta(s, \theta) &= -\theta + s \int^{\theta} J R^{-2} d\theta \\ f(s) &= 2\pi \left(s \int^{\theta} J R^{-2} d\theta \right)^{-1}. \end{aligned}$$

Since the toroidal angle ζ is ϕ in the VMEC code, δ is small even in the averaged equilibrium geometry. Finally, a toroidal current consistent with the stellarator expansion is given by

$$R_{J\theta}(s, \theta) = -p'(s) \Omega(s, \theta) + G(s).$$

In the case of a net current free equilibrium, where the flux surface averaged plasma current is zero,

$$G(s) = p'(s) \oint \Omega(s, \theta) R^{-2} J d\theta / \oint R^{-2} J d\theta.$$

These averaged equilibrium quantities are passed to the STEP stability code. In this code, the eigenvalues and eigenfunctions of global low- n modes (n is a toroidal mode number) are

obtained by calculating the extrema of the Lagrangian

$$\begin{aligned} \mathcal{L} &= 2(K - \delta W) \\ &= \omega^2 \int \rho |\xi_1|^2 d\tau \\ &\quad - \int (|Q_1|^2 + j_{||} \times \xi_1^* Q_1 + (R_0/R)^2 (\xi_1 \cdot \nabla p)(\xi_1^* \cdot \nabla \Omega)) d\tau, \end{aligned}$$

where

$$\begin{aligned} \xi_1 &= \xi_1^{(0)} = R^2 \nabla \phi \times \nabla \eta, \\ Q_1 &= Q_1^{(0)} = \nabla \phi \times \nabla (R^2 B_{z,II} \cdot \nabla \eta), \end{aligned}$$

and η is a stream function giving the perturbation. In the derivation of the Lagrangian, we used the usual scalar expansion ordering, but kept the finite toroidal curvature effect in the geometrical factor. The numerical method and the solver are the same as those in the PLST2 tokamak stability code.

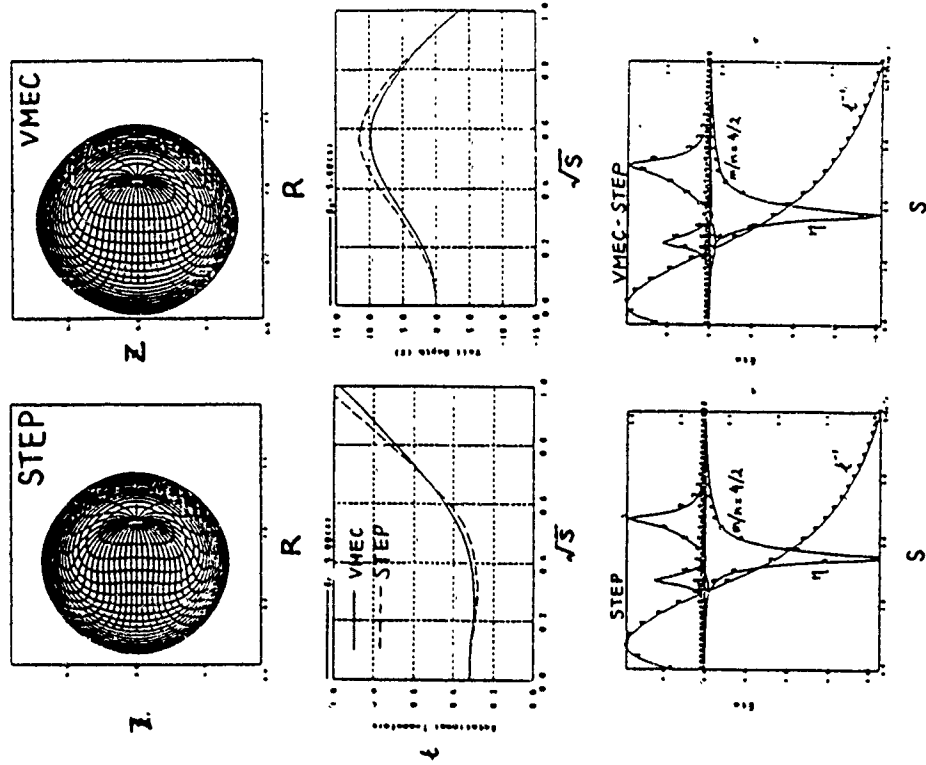
The figure shows averaged flux surfaces, the rotational transform profiles, the well depth, parameter profiles, and the mode structure of $n = 2$ linear perturbation from equilibrium for an ATF toratron⁴ at a central beta of 5% when the vacuum magnetic axis is shifted inwards by 10 cm from the standard position. These equilibria were obtained from the 2-D STEP code and the 3-D VMEC code with the VMEC results averaged over ζ . The shapes of the averaged flux surfaces are slightly different; however, good agreement is obtained for the rotational transform and the well depth parameter profiles. The stability analysis results are also in good agreement.

In summary, a difference in geometrical factors of the averaged flux surfaces does not affect the stability significantly, but a difference of the s and V' profiles does, since they are amplified through the position of the mode resonant surface and the pressure gradient. We think that the VMEC-STEP code is useful for study of the stability of a heliotron/toratron with a low aspect ratio or helical pitch number, since the VMEC code gives realistic three-dimensional finite beta equilibrium data such as the position of the resonant surface, magnetic shear, and local average curvature to the STEP stability solver. It will be useful to extend this study to compare stability calculations using STEP and the new low- n stability code TERPSICHOE.⁵

REFERENCES

1. Y. Nakamura, K. Ichiguchi, M. Wakatani, and J. L. Johnson, *J. Phys. Soc. Japan* **58**, 3157 (1989).

2. S. P. Hirshman, W. I. van Rij, and P. Merkel, *Comput. Phys. Commun.* **43**, 143 (1986).
3. G. Anania and J. L. Johnson, *Phys. Fluids* **26**, 3070 (1983).
4. J. F. Lyon, G. L. Bell, J. D. Bell *et al.*, *Fusion Tech.* **17**, 33 (1990).
5. D. V. Anderson, W. A. Cooper, R. Gruber *et al.*, *Scientific Computing on Supercomputers II* (Plenum Press, New York, 1990) p.159.



Towards a Realistic Plasma Simulation Code¹

David V. Anderson
Lawrence Livermore National Laboratory
Livermore, CA 94550

Introduction

Several new developments in the technology of simulating plasmas, both in particle and fluid models, now allow a stage of synthesis in which many of these advances can be combined into one simulation model. Accuracy and efficiency are the criteria to be satisfied in this quest. We want to build on the following research:

- 1 The development of the δf method of Barnes[4]
- 2 The moving node Galerkin model of Glasser, Miller and Carlson[5]
- 3 Particle moving schemes on unstructured grids by Ambrosiano and Brandon[1]
- 4 Particle simulations using sorted particles of Anderson and Shumaner[3]

Rather than being competing developments, these presumably can be combined into one computational model.

We begin by summarizing the physics model for the plasma. The Vlasov equation can be solved as an initial value problem by integrating the plasma distribution function forward in time.

Tracer Particle Truncation

Two principal approaches have been taken in these simulations: following fluids or following particles. The former suffer from incorrect truncation of the moment equations, while the latter produce large non-physical fluctuations in the macroscopic variables. Many years

¹This work was supported by the U. S. Department of Energy under contract W-7408-ENG-46 with the Lawrence Livermore National Laboratory

ago, an effort was made to truncate moments of the Vlasov equation by using a sparse set of simulation tracer particles to provide the higher moment (in that case the heat flow tensor.)[2] The method was shown to work correctly and kinetic phenomena, such as Landau damping, was simulated. However, it was learned from studies of the method's accuracy that the number of tracer particles required was unacceptably large and, in fact, was of the order of the number of simulation particles required in a standard particle simulation of the same plasma. Thus, the method was abandoned.

The δf Method

The δf method also uses particles to truncate the moment equations. It proceeds by evolving both the appropriate fluid moments as well as particles which sample a non-Maxwellian partition of the total distribution function. This is done in such a way that the fluid equations are properly truncated from the stress tensor and heat flow tensor obtained from the particles. The principal benefit of this model is a very large reduction in numerical fluctuations of the density and other macroscopic fields. When and where the plasma is mostly describable as a fluid, the moment equations are dominant in determining the evolution. Alternatively, where the plasma is wildly kinetic, with a vanishing Maxwellian component, the particle simulation treatment dominates. And where the plasma distribution has important contributions from both the Maxwellian component and the δf component, a correct combination of the two models is used.

Moving Finite Elements

Having adopted such a model does not preclude using the other techniques alluded to above. The fluid evolution can be treated by the MFPA (moving finite element algorithm) which is derived from a Galerkin formulation of the fluid equations.[5] Rather than minimize the error integral with respect to the basis function amplitude coefficients, a more general minimization is done with respect to both these amplitude coefficients and the nodal positions. With these

additional degrees of freedom the magnitude of the error integral is reduced dramatically. In practice one can use a much coarser grid than what is conventionally used and still enjoy a more accurate evolution of the fluid equations.

Particle Simulation on Unstructured Grids

If the best fluid algorithm dictates using a moving grid, we would like to perform the particle kinematics on the same grid. Since the MFEA method actually employs a triangular unstructured grid, it is natural to employ the same for the particles. Such a method has been recently developed for plasma simulations of devices (such as diodes) with complicated domains [1]. A first approach would be to use the grid dictated by the MFEA method.

Parallelization via Ordered Particles

Lastly, we are developing algorithms that will allow these simulations to be run on massively parallel computers. Our efforts have been concentrated on constructing particle codes that use ordered particles. They are kept ordered by sorting algorithms tailored to work efficiently on slightly disordered distributions. The HOPS code (Hybrid Ordered Particle Simulation) uses these techniques and has been successfully run and shown to be efficient in most respects [3]. Its particle sorting and reordering scheme is only marginally efficient and is currently being optimized by developing algorithms that run in a number of steps linear in the number of particles to be sorted.

Summary

It seems that we have arrived at a moment where significant improvements can be made in the computational technology for simulating plasma phenomena. Several developments, cited here, seem to lay the groundwork for a new class of plasma simulation codes that will be more faithful to the theory, more precise, and which will be efficiently computed in massively parallel computing environments.

To these ends we are planning to develop the HOPS code as follows:

- 1 Install linear sorting algorithm
- 2 Parallelize by domain decomposition for vector parallel machines
- 3 Convert to unstructured grids
- 4 Add MFEA fluid code and δf interface
- 5 Generalize MFEA Galerkin treatment to include δf contributions

References

- [1] John Ambrosiano and Scott Brundson, *Finite Element Particle Simulation on Unstructured Grids*, Proc. Physics Computing '91, San Jose, (1991)
- [2] D. V. Anderson, *Collisionless Plasma Fluid Model Truncated by Tracer Particles*, UCRL Report 76750, Lawrence Livermore National Laboratory, (1975); Proc. 7th Conf. on the Numerical Simulation of Plasmas, New York City, (1975)
- [3] David V. Anderson, Bruce C. Curtis, Dana E. Shumann, Eric J. Horowitz, *Obtaining Gigaflop Performance from Particle Simulation of Plasmas*, Proc. Gray Science and Engineering Symposium, London, October (1990)
- [4] D. C. Barnes, *A New Implicit δf Method for Plasma Simulation*, Proc. Sherwood Theory Meeting, Seattle, (1991)
- [5] Alan H. Glasser, Keith Miller, and Neil Carlson, *MFEA: A General-Purpose 2D Moving Finite Element Code for Nonlinear, Time-Dependent Fluid Problems*, Proc. Physics Computing '91, San Jose, (1991)

ELECTROMAGNETIC MODELLING IN ARBITRARY GEOMETRIES BY THE VIRTUAL PARTICLE PARTICLE-MESH METHOD

by

James W Eastwood and Wayne Arter
AEA Technology, Culham Laboratory, Abingdon, Oxon, OX14 3DB,
England

1 INTRODUCTION

Virtual particle (VP) particle-mesh algorithms are a new type of particle method for solving the relativistic Vlasov-Maxwell equations. Unlike conventional Particle-In-Cell (PIC) schemes [1], they are derived using finite elements in both space and time. Current is assigned from "virtual particles" placed at points specially interpolated between positions at successive timelevels; charge conservation is automatic.

The advantages of VP have been discussed elsewhere [2], and 2-D VP electromagnetic codes have been successfully employed to simulate microwave sources [3]. Refs [3] used uniformly distributed finite element nodes in an axially symmetric geometry. Only a restricted class of device is well modelled in such circumstances, generating a need to implement VP in more complex geometries. In this paper we focus on the development of a solver for Maxwell's Equations that is compatible with VP in general curvilinear coordinates.

2 VARIATIONAL FORMULATION

In general curvilinear co ordinates $(\hat{x}^1, \hat{x}^2, \hat{x}^3)$ the electromagnetic field action integrand may be written as

$$I = \int dt d\hat{x}^1 d\hat{x}^2 d\hat{x}^3 \sqrt{g} \left\{ \frac{1}{2} (E, D^1 - H, D^2 + J^1 A, -\rho\phi) \right\} \quad (1)$$

where

$$E_k = -\frac{\partial\phi}{\partial\hat{x}^k} - A_k \quad (2)$$

$$B^1 = \frac{e^{ijk} \partial A_k}{\sqrt{g} \partial \hat{x}^j} \quad (3)$$

$$D^i = e_i E^i = e_i g^{ij} E_j \quad (4)$$

$$H_i = \frac{B_i}{\mu_0} = \frac{1}{\mu_0} g_{ij} B^j \quad (5)$$

$$J^i = \sum_p \frac{q_p}{\sqrt{g}} \delta(\hat{x}^1 - \hat{x}_p^1) \delta(\hat{x}^2 - \hat{x}_p^2) \delta(\hat{x}^3 - \hat{x}_p^3) \dot{\hat{x}}^i \quad (6)$$

$$\rho = \sum_p \frac{q_p}{\sqrt{g}} \delta(\hat{x}^1 - \hat{x}_p^1) \delta(\hat{x}^2 - \hat{x}_p^2) \delta(\hat{x}^3 - \hat{x}_p^3) \quad (7)$$

The sum over p is over particles, each with charge q_p . The metric tensor elements g_{ij} can be computed from the relationship between Cartesian and the general curvilinear coordinates i.e. given

$$\hat{x}^i = \hat{x}^i(\hat{x}^1, \hat{x}^2, \hat{x}^3), \quad (8)$$

then

$$g_{ij} = \frac{\partial \hat{x}^a}{\partial \hat{x}^i} \frac{\partial \hat{x}^a}{\partial \hat{x}^j}, \quad (9)$$

$$g = |g_{ij}|, \quad (10)$$

and

$$g_{ij} g^{jk} = \delta_i^k. \quad (11)$$

Treating I as a functional of the vector potential A_i , the scalar potential ϕ and particle co-ordinates $\{\hat{x}_p\}$ leads to Euler Lagrange equations representing Maxwell's equations and relativistic particle motion.

Substituting test function approximations for ϕ , A_i and \hat{x}_p and varying with respect to the nodal amplitudes yields a finite element approximation to the Maxwell-Vlasov equations. Now, if we introduce $\hat{v} = \sqrt{g} D^i$, $\hat{d}^i = \frac{1}{\sqrt{g}} D^i$, $J^i = \sqrt{g} J^i$ and $Q = \sqrt{g} \rho$, Maxwell's equations become

$$\frac{\partial \hat{v}^i}{\partial t} = -e^{ijk} \frac{\partial E_k}{\partial \hat{x}^j}, \quad \frac{\partial \hat{v}^i}{\partial \hat{x}^j} = 0, \quad (12)$$

$$\frac{\partial \hat{d}^i}{\partial t} = e^{ijk} \frac{\partial H_k}{\partial \hat{x}^j} - J^i, \quad \frac{\partial \hat{d}^i}{\partial \hat{x}^j} = Q, \quad (13)$$

where e^{ijk} is the permutation symbol. This suggests, and our analysis confirms, that the discrete VP equations in the quantities \hat{v}^i , \hat{d}^i , E_i , H_i , J^i and Q are identical in any co-ordinate system. Geometrical information appears, along with the permeability and permittivity tensors, only in the constitutive relations relating H to B and D to E .

3 GENERATING FINITE ELEMENTS

We treat the Vlasov-Maxwell equations in a general curvilinear co ordinate system because these arise naturally when we seek to represent awkwardly shaped devices. Transfinite interpolation [4] provides a co-ordinate system for any simply connected surface (in 2-D) or volume (in 3-D) provided the bounding curves or surfaces are not too convex or concave. The volume is regarded as a hexahedron: six "faces" and eight "corners" $\{\hat{x}_{ijk} : i = 0, 1; j = 0, 1; k = 0, 1\}$ are defined by twelve lines on its boundary constituting the "edges". Let $X_i(s, t)$, $s, t \in [0, 1]$ represent opposite surfaces passing through the four points \hat{x}_{0jk} for $i = 0$ and \hat{x}_{1jk} for $i = 1$; $Y_j(r, t)$ and $Z_k(r, s)$ similarly represent the remaining "faces". Further if $X_{ij}(r)$, $r \in [0, 1]$ are the four edges

joining corner node x_{0j} to x_{1j} , each i and j , with $Y_{ij}(s)$ and $Z_{ij}(t)$ defined similarly, then the transfinite interpolant is

$$\begin{aligned} x(r, s, t) = & X_1(s, t)\phi_1(r) + Y_2(r, t)\psi_2(s) + Z_3(r, s)\eta_3(t) \\ & - X_2(r)\psi_2(s)\eta_3(t) - Y_3(s)\phi_3(r)\eta_3(t) - Z_{ij}(t)\phi_i(r)\psi_j(s) \\ & + x_{ijk}\phi_i(r)\psi_j(s)\eta_k(t), \end{aligned} \quad (14)$$

where the summation convention is implied and ϕ_i , ψ_j and η_k are cardinal basis functions

If one or more of the surfaces is not given in this form, but only in terms of its bounding curves, then we use the two dimensional transfinite interpolant to give an expression that may be substituted in Eq (14) r, s and t are to be identified with x^1, x^2 and x^3 respectively.

Realistic microwave sources involve cavities and waveguides of cylindrical and rectangular cross section. Separate subdomains are therefore introduced to handle the various components and also axial effects. Figure 2 shows the multiblock representation of a cross-section through a cylindrical device in a plane where a waveguide is attached.

4 FINAL REMARKS

The combination of a general curvilinear finite element formulation and multiblock decomposition of complex regions into logical ijk blocks makes accurate particle simulation practicable in complex geometries. Multidimensional electromagnetic PIC codes have been in use for many years, but almost all mature codes have been limited to regular orthogonal meshes, where curved surfaces have been crudely treated in a stepwise fashion. The element oriented formulation and the charge conserving property of the VP algorithms make them well suited for parallel implementation on both SIMD and MIMD architecture computers. These properties, combined with the power of modern workstations, offer the prospect of desktop particle simulation CAD capabilities to the microwave engineer.

5 REFERENCES

- [1] R W Hockney and J W Eastwood, "Computer Simulation using Particles", McGraw-Hill (1980) (student edn. Adam-Hilger, 1988)
- [2] J W Eastwood, Comput Phys Commun. 64, 252 (1991)
- [3] J W Eastwood, paper 3D1, IEEE Int. Conf Plasma Sci., Buffalo (May, 1989), J W Eastwood and T C Hender, paper IM1 13th Conf. Num. Sim. Plasma, Santa Fe (September 1989).
- [4] W J Gordon and C A Hall, Int J. Numer. Meths. Engng. 7, 461 (1973)

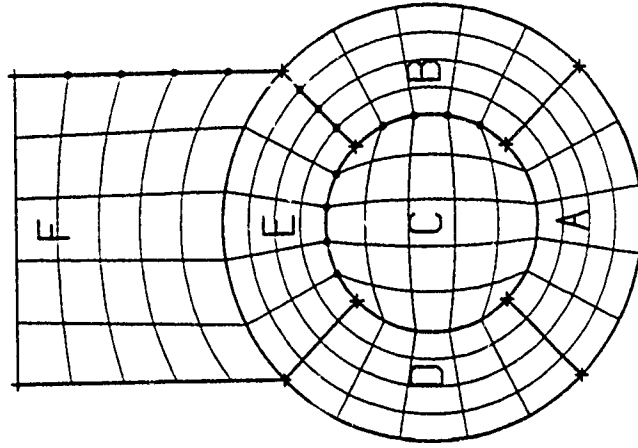


Figure 1: This shows in cross section the division of a model device into six multiblocks A-F. Crosses mark the corners of the blocks and the thicker lines are their boundaries

Solutions to Poisson's Equation on a 3D Rectangular Mesh with Internal Conductors

David P. Grote, Alex Friedman, and A. Bruce Langdon
Lawrence Livermore National Laboratory

I. Haber
Naval Research Laboratory

Introduction

The WARP code^{1,2,3} is used for modeling the transport of space charge dominated heavy ion beams with applications to heavy ion fusion. The beam induces significant charges on the surrounding pipe and on any conductors within the pipe, and this in turn affects the behavior of the beam. Electrostatic quadrupole focusing requires the presence of conductors within the pipe, see figure 1. Additional charges on these conductors produce the focusing fields. In a simulation model, these charges are not known initially, but are found by forcing the conductors to be at the required voltage during the field solution. A fully consistent solution to Poisson's equation is required, including the beam space charge, any induced charges, and the focusing fields. This field is required at every time step, so the solution method must also be fast. The methods we have examined include capacity matrix methods⁴ and a new method that finds the induced charge iteratively. The methods are based on a 3D FFT (sine-sine-periodic, in x , y , z , respectively) solution of Poisson's equation with no internal conductors. Here, x and y are the transverse coordinates, and z is the axial coordinate. We are motivated by a desire to compare our simulations to the MBE-4 experiment at LBL⁵ which uses electrostatic quadrupole focusing.

Capacity Matrix Method

The capacity matrix method is direct (non-iterative), but suffers from size problems in 3D. For example, a typical case might use a grid with dimensions $64 \times 64 \times 128$ and have seven focusing elements, each eight grid cells long in z . About 64 points are needed in each x - y plane where a focusing element exists, 16 points per conductor per z -slice. This number is an estimate of the number of points needed for good field accuracy in an absolute sense, and is based on experience with a similar method for axially-independent conductors.⁶ The total number of points is then 3584. Including all of the conductor points in a single matrix would require a 3584×3584 full matrix, which is prohibitively large. The matrix can be broken down by assuming that the conductors act independently, that is, are widely separated. The focusing lens consists of four identical conductors. Each can be represented by the same matrix even though they may be held at different voltages. With seven focusing elements, there are 28 separate conductors of 128 points each. This reduces the matrix to a 128×128 full matrix. Since couplings between the conductors are neglected, this only gives an approximate solution.

It may be possible to include a number on points from each of the other conductors as well into each "local" matrix. Since they are physically distant, misrepresenting their shapes may not lead to serious errors. Only the induced charge on the element in question

would actually be used, since that on the other elements would be handled by their own capacity matrices.

Simple Iterative Method

The new method is an iterative method; it is related to a method for resistive blocks described earlier⁷. At each iteration, the locally correct induced charge, ρ_{induced} , is calculated at the conductor surface via $\rho_{\text{induced}} = \nabla \cdot \mathbf{E}$, where an improved electric field, \mathbf{E} , has been obtained as described below by setting the conductors to the desired voltages. The new fields are then solved for with the 3D FFT, including the induced charge on the conductors. The iterations converge to the globally correct solution.

In 3D, to save space, the electric fields are never explicitly calculated; only the electrostatic potential, Φ , is known. To calculate ρ_{induced} , we use

$$\rho_{\text{induced}} = \nabla \cdot \mathbf{E} = -\nabla^2 \Phi.$$

The usual seven point finite difference is used to calculate $\nabla^2 \Phi$. When the finite difference is calculated, the potentials on the conductor surfaces and in their interiors are set to the desired voltage. This reduces the finite difference to the difference between the desired voltage and the potential just outside the conductor divided by Δ^2 , where Δ is the grid size in the appropriate direction. This is equivalent to the expression for the induced surface charge,

$$\sigma_{\text{induced}} = \Delta \rho_{\text{induced}} = |V_{\text{desired}} - \Phi|/\Delta = |E_{\text{normal}}|,$$

where E_{normal} is the normal electric field at the conductor surface.

This method is fairly slow, requiring many iterations for convergence. It is desirable to reduce the number of iterations, since each iteration includes a 3D FFT Poisson solve. Several methods have been tried to accomplish this.

The iterations, which in some cases oscillate, can be underrelaxed with the following relation,

$$\rho^{n+1} = \alpha \bar{\rho}^{n+1} + (1 - \alpha) \rho^n$$

where α is the underrelaxation parameter and is between zero and one, and $\bar{\rho}^{n+1}$ is the result obtained without underrelaxation. Experimentation has shown that the optimum value for α is about .35 in a typical case. The second column in Table 1 shows the rate of convergence with $\alpha = .35$.

The number of iterations can be decreased by use of a good initial guess. The induced charges from the previous time step can be used. This only reduces the number of iterations by a few, however. The third column in Table 1 shows the rate of convergence.

The capacity matrix solution that uses a separate matrix for each conductor can also be used as an initial guess. The iteration should correct for any dependencies that the separate conductors may have on each other. We plan to test this in the near future.

Summary and Plans

So far the best results with the simple iteration have been obtained using a fixed underrelaxation parameter value of .35. Initial results using the Steffenson acceleration method⁶ suggest that it will not help much. We believe that the capacity matrix solution which uses a separate matrix for each conductor followed by iteration will be effective. We also plan to look into other schemes, including line-relaxation methods and variants of the alternating direction implicit method.

References

1. A. Friedman, D. A. Callahan, D. P. Grote, A. B. Langdon, and I. Haber, *Proc. of Conf on Computer Codes and the Linear Accelerator Community*, Los Alamos, NM January 21-25, 1990, R. K. Cooper and K. C. D. Chan, Eds.
2. A. Friedman et. al., this conference
3. D. A. Callahan et. al., this conference
4. R. W. Hockney and J. W. Eastwood, *Computer Simulations Using Particles*, Adam Hilger, Bristol, 1988, p. 215.
5. S. Eylon et. al., "Drift Compression Experiments on MBE-4 and Related Emittance Growth Phenomena", *Proc. of the Fourteenth Biennial IEEE/APS Particle Accelerator Conf.*, San Francisco, CA, May 6-9, 1991
6. D. P. Grote et. al., "3D Simulations of Axially Confined Heavy Ion Beams in Round and Square Pipes", *Proc. of the International Symposium on Heavy Ion Inertial Fusion*, Monterey, CA, December 3-6, 1990
7. D. E. Nielsen, J. J. Ambrosiano, A. Friedman, and A. B. Langdon, "Resistive Block Methods for Electrostatic Field Solutions with Internal Elements" (final two authors added after submission of abstract), *Proc. 13th Conf. on the Numerical Simulation of Plasmas*, Santa Fe, NM, 1989
8. S. D. Conte and C. de Boer, *Elementary Numerical Analysis: An Algorithmic Approach* (McGraw-Hill, New York, 1972), pp 50-57.

Table 1

number of accurate digits†	number of iterations $\alpha = .35$	good guess
2	5	4
4	16	14
6	27	24
8	36	35
10	48	45

†Accuracy determined by rms error of potential on conductor

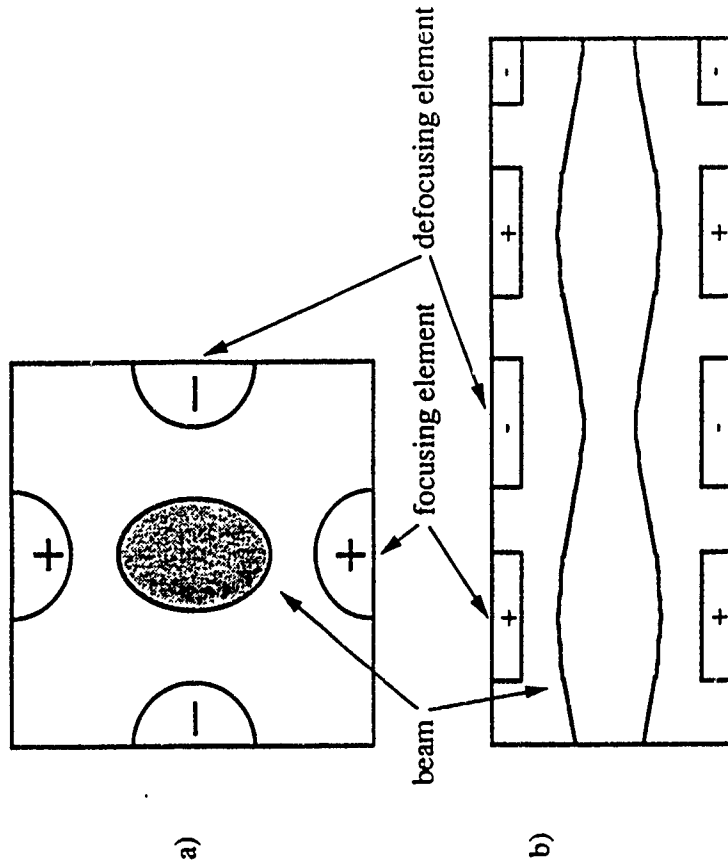


Figure 1. Geometry of electrostatic quadrupole focusing lens.

a) slice at fixed z for quadrupole which is focusing in vertical direction y, defocusing in horizontal direction x; b) slice at fixed x.

NUMERICAL METHODS FOR DETERMINATION OF THE CHARGE
PARTICLE ADIABATIC MOTION REGION IN MAGNETIC FIELDS

S.V. Kuzmin

630090, USSR, Novosibirsk, Institute of Nuclear Physics

ABSTRACT

The problem of determination of the charge particle adiabatic motion region in different types of magnetic field is investigated by means of numerical methods developed from different stochastic motion criteria.

Three methods are studied. The first one is the traditional method based on study of magnetic moment diffusion. The second one is the method of nearest particle trajectory. The third one is the semi-analytical method based on numerical determination of fundamental parameters of particle motion.

These three methods are applied for different types of magnetic fields. The specific features of each method are analyzed. The advantages and drawbacks of the methods are discussed

CELESTID: An Implicit, Fully Kinetic Model for Low Frequency, Electromagnetic Plasma Simulation

H. X. Vu and J. U. Brackbill
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

An implicit, one-dimensional, fully electromagnetic particle simulation code (CELESTID) is presented in which both electrons and ions are treated kinetically using finite-size particles. The implicit time-advanced field equations are cast in a form convenient for Gaussian elimination, and the field equations are solved directly for arbitrary time steps without iterations. CELESTID has been tested on a number of standard problems, and the results are similar to those obtained by existing one-dimensional electromagnetic particle and hybrid codes. Results from test simulations of the two stream instability and of the electromagnetic, low-density ion beam instability are presented. Perpendicular fast shocks and oblique switch-off shocks, physical mechanisms relevant to the conversion of magnetic energy to solar wind particle kinetic energy at the earth's bow shock, have also been simulated using CELESTID by reflecting both electrons and ions specularly from a physical piston. CELESTID yields the correct shock speeds for, and the correct Rankine-Hugoniot jump conditions across, both types of shocks. Because of limited space, the features of the shocks are not discussed in this extended abstract, but will be discussed in detail at the time of presentation.

1. INTRODUCTION

The essence of particle simulations is to model a plasma as a collection of individual particles. The individual particle motion must then conform to Newton's laws. The electromagnetic forces affecting the motions of individual particles are usually modeled as self-consistent fields, i.e., electromagnetic fields which are generated by the particles' collective motion.

There is a large number of numerical algorithms available for solving the system of coupled field-particle equations, e.g., *time-explicit* and *time-implicit* algorithms (cf. [1-17]). Time-explicit algorithms use the electromagnetic forces at the current time level to advance the particles' velocities and positions, and are thus relatively simple to implement. Unfortunately, numerical stability is achieved only if the time step of the simulation is small enough to resolve the fastest time scale (usually the electron time scale), making it impractical to simulate low-frequency phenomena using time-explicit algorithms. Time-implicit algorithms, on the other hand, use the electromagnetic forces at some intermediate time level to advance the particles' velocities and positions. Numerical stability for time-implicit algorithms is achieved if the time step of the simulation is small enough that the particles do not traverse more than one cell in a time step. This requirement is much less stringent than that for time-explicit algorithms. In fact, time-implicit algorithms often allow time steps several orders of magnitude larger than those allowed by time-explicit algorithms. Consequently, even though time-implicit algorithms are more difficult to implement and more expensive to execute (per time step) than time-explicit algorithms, they often offer substantial overall savings in computer time when the problems of interest involve low frequency phenomena.

CELESTID, an implicit, one dimensional, fully electromagnetic particle simulation code is described. Electrons and ions are treated kinetically using finite-size particles. CELESTID incorporates a variation of the implicit moment method proposed and implemented by Brackbill and Forslund in the VENUS code [6-7]. The principal differences between CELESTID and VENUS are (1) the moment expansions, and (2) the formulation of the time-advanced field equations. In VENUS, the time-advanced field equations are solved iteratively. Because of the limitation of the iterative algorithm used in solving the field equations, one could not realize the full advantage of

the implicit moment method with VENUS, i.e., the maximum allowable time step is restricted by the iterative field solver, not by the implicit moment algorithm. In CELESTID, we seek what has been possible until now only in principle. Namely, we seek to solve the plasma simulation equations for arbitrary time steps so that problems with multiple time scales can be simulated efficiently. To achieve this goal, the implicit moment equations are reformulated so that the time-advanced field equations are solved directly by Gaussian elimination. As a result, one can achieve arbitrarily large time steps (not possible with VENUS), subject only to an accuracy constraint on the moment expansion.

II. FORMULATION

The motion of a charged particle in an electromagnetic field is described by:

$$\left. \begin{aligned} \frac{dx}{dt} &= u \\ \frac{du}{dt} &= \frac{q}{m} \left(E + \frac{u \times B}{c} \right) \end{aligned} \right\} \quad (1)$$

The electromagnetic field influencing the particle motion is described by Maxwell's equations. For convenience, the E and B fields are formulated in terms of the scalar and vector potentials using the Coulomb gauge:

$$\left. \begin{aligned} \frac{1}{c^2} \frac{\partial^2 A}{\partial t^2} - \nabla^2 A &= \frac{4\pi}{c} \mathbf{j} - \frac{1}{c} \frac{\partial \nabla \phi}{\partial t} \\ \nabla^2 \phi &= -4\pi \rho \\ \mathbf{B} &= \nabla \times \mathbf{A} \\ \mathbf{E} &= -\nabla \phi - \frac{1}{c} \frac{\partial \mathbf{A}}{\partial t} \end{aligned} \right\} \quad (2)$$

where $\nabla \cdot \mathbf{A} = 0$, \mathbf{b} is the external magnetostatic, uniform field.

Using the time implicit differencing scheme and the moment expansions about the particles' initial positions [15,18], Eqs. (1)-(2) can be spatially and temporally discretized, and the resulting field-particle coupled equations are solved in order to advance the fields as well as the particles' velocities and positions.

The implicit moment method can be implemented in three stages, all of which can be done efficiently. The first stage involves solving for the scalar potential at the current time level from the actual particle data. This step is taken to help ensure that $\nabla \cdot \mathbf{A} = 0$. The second stage involves solving for the intermediate scalar potential and the time-advanced vector potential using the moment expansions. The third stage involves using the temporally discretized Newton's laws and the estimated fields obtained from the second stage to advance the particles' velocities and positions.

III. NUMERICAL EXAMPLES

Results from the simulations of the two-stream instability and of the electromagnetic, low-density ion beam instability using CELESTID are presented.

A. TWO-STREAM INSTABILITY

The instability is caused by counter-streaming electrons with $u_0/c = \pm 0.10x$. The simulation is performed in the frame of the neutralizing background ions. The ion-electron mass ratio used in the simulation is $m_i/m_e = 100$. The simulation box is periodic, and has a physical length $L = c/\omega_{pe}$. The electrons and ions are taken to have the same temperature with $\lambda_{De} = 10^{-1} c/\omega_{pe}$. The simulation box has 32 cells of equal size. Each species is represented by 10 particles per cell. The maximum allowable time step is $\omega_{pe} \delta t_m \approx 0.3$. For this simulation run, $\omega_{pe} \delta t = 0.15$. The time-decentering parameter is $\theta = 1/2$. The results are shown in Fig. 1 (time is in units of ω_{pe}^{-1})

B. ELECTROMAGNETIC, LOW-DENSITY ION BEAM INSTABILITY

The instability is caused by an ion beam streaming through a much denser, magnetized background plasma ($u_i/v_A = 10$, $n_i/n_0 = 1.5\%$). The simulation is performed in the frame of the background ions. The ion-electron mass ratio used in the simulation is $m_i/m_e = 100$. The simulation box is periodic, and has a physical length $L = 256c/\omega_{pe}$. The electrons, background ions, and beam ions are taken to have the same temperature with $\beta_e = \beta_i = \beta_b = 1$. The external magnetostatic, uniform field $b = b\hat{x}$ is chosen so that $\omega_{ci}/\omega_{pe} = 10^4$. The simulation box has 128 cells of equal size. Each species is represented by 10 particles per cell. The maximum allowable time step is $\omega_{pe}\delta t_m \approx 0.2$. For this simulation run, $\omega_{pe}\delta t = 0.025$. The time-decentering parameter is $\theta = 0.55$. The results are shown in Fig. 2 (time is in units of ω_{pe}^{-1}).

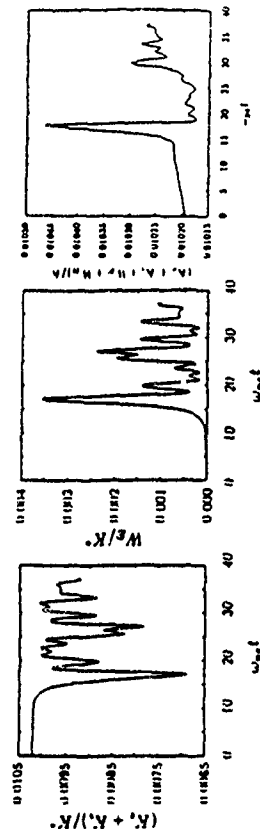
IV. CONCLUSIONS

The results shown in Figure 1 for the two-stream instability show the correct qualitative behavior (thermalization, energy exchange between the electric field energy and the particle energy, etc.). The error in the total energy of the system is about 10% of the maximum electrostatic field energy. The results for the electromagnetic, low-density ion beam instability shown in Figure 2 compare favorably with the results from hybrid codes [19]. The energy exchange between the ion beam and the magnetic field, the relative magnitude of positive-helicity and negative-helicity waves, and the direction of the dominant wave component, show the correct qualitative behavior [10].

The test problems cover a wide range of temporal and spatial scales. CELESTID performs well for both large and small time steps. For the same simulation parameters, CELESTID takes about twice as long (CPU time) as a typical hybrid code [20].

V. REFERENCES

1. D. Dickman, R. L. Mott and C. W. Nielson, *Phys. Fluids* **12**, 1708 (1969).
2. A. B. Langdon, *J. Comput. Phys.* **6**, 217 (1970).
3. A. B. Langdon and B. F. Lasinski, *Methods Comput. Phys.* **16**, 327 (1976).
4. C. W. Nielson and H. R. Lewis, *Methods Comput. Phys.* **16**, 367 (1976).
5. J. Busnardo-Neto, P. L. Pritchett, A. T. Lin and J. M. Dawson, *J. Comput. Phys.* **23**, 300 (1977).
6. J. A. Byers, B. I. Cohen, W. C. Condit and J. D. Hanson, *J. Comput. Phys.* **27**, 363 (1978).
7. D. W. Hewett and C. W. Nielson, *J. Comput. Phys.* **29**, 219 (1978).
8. D. W. Hewett, *J. Comput. Phys.* **38**, 378 (1980).
9. R. J. Mason, *J. Comput. Phys.* **41**, 233 (1981).
10. J. Denavit, *J. Comput. Phys.* **42**, 337 (1981).
11. J. U. Brackbill and D. W. Forslund, *J. Comput. Phys.* **46**, 271 (1982).
12. D. W. Forslund and J. U. Brackbill, *Phys. Rev. Lett.* **48**, 1614 (1982).
13. J. M. Dawson, *Rev. Mod. Phys.* **55**, 403 (1983).
14. R. J. Mason, in *Multiple Time Scales*, edited by J. U. Brackbill and B. I. Cohen (Academic Press, New York, 1985), Chap. 8.
15. J. U. Brackbill and D. W. Forslund, in *Multiple Time Scales*, edited by J. U. Brackbill and B. I. Cohen (Academic Press, New York, 1985), Chap. 9.
16. B. I. Cohen, in *Multiple Time Scales*, edited by J. U. Brackbill and B. I. Cohen (Academic Press, New York, 1985), Chap. 10.
17. A. B. Langdon and D. C. Barnes, in *Multiple Time Scales*, edited by J. U. Brackbill and B. I. Cohen (Academic Press, New York, 1985), Chap. 11.
18. H. X. Vu and J. U. Brackbill, *Comput. Phys. Commun.*, submitted for publication.
19. D. Winske and M. M. Leroy, *J. Geophys. Res.* **89**, 2673 (1984).
20. D. Winske, private communication.



$$K^* = \left[(n_e + n_i) \frac{c}{\omega_{pe}} \right] \frac{1}{2} m_e c^2$$

Figure 1: Two-stream instability. Kinetic energy (left), electric field energy (middle), and total energy (right). The magnetic field energy is many orders of magnitude smaller than the electric field energy, and is not shown.

$$\frac{n_b}{n_b + n_i} = 1.5\%$$

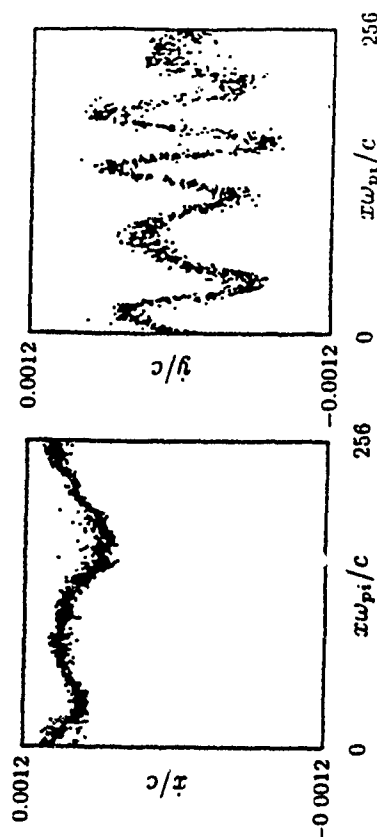


Figure 2: Electromagnetic, low-density ion beam: instability. Phase-space plots of ion beam at $\omega_{pe} t = 20$. The slowing down of the ion beam in the longitudinal direction ($x-z$) and the coherent modulation of the ion beam's transverse velocity ($x-y$) are evident. The positive helicity, $B_y^+(z, t)$, and negative-helicity, $B_y^-(z, t)$, components of $B_y(z, t)$ (not shown) indicate that the positive-helicity component accounts for almost all of $B_y(z, t)$, and the mode propagates to the right corresponding to the resonant case.

A High-Order Moment Simulation Model*

K T Tsang, C Kostas, D P Chernin, J. J. Petillo, and A. Mondelli
Science Applications International Corporation
1710 Goodridge Dr., McLean, VA 22102

1. Introduction

We consider a beam transport system in which the particle motion in the two transverse directions (x and y) are coupled. Chernin¹ has studied the beam envelope, with linear transverse spatial dependence of magnetic field, for a mono-energetic particle beam. Here, we are interested in a beam, with a distribution in energy, in a transport system with nonlinear transverse dependence of the magnetic field. The spatial coordinate (z) along the beam motion can no longer be equated to the time variable. Fast particles will overtake the slower particles as time evolves. Therefore, the beam is described by a set of partial differential moment equations, instead of a set of ordinary differential envelope equations. This aspect of our approach distinguishes it from the work of Chernin¹ and of Channell and coworkers^{2,3}. The theory by Channell et al. uses moment equations to model a three-dimensional beam bunch. For long bunches, however, it is impractical to carry high enough longitudinal moments to model the oscillations within the bunch. In the following we will derive a relativistically covariant form of moment equations, based on the work of Newcomb⁴ and Amendt and Weitzner⁵.

2. Relativistic Formulation

We start with a time coordinate t and local Cartesian space coordinate (x^1, x^2, x^3), replacing the usual coordinate (x, y, z), where x^3 is measured along the beam motion direction and x^1 and x^2 are the transverse directions. We define $x^4 = ct$, where c is the speed of light, so that spacetime is parametrized by $x^\mu, \mu = 1, 2, 3, 4$. We use a summation convention, and we assume that Latin subscripts and superscripts, i, j, k, l , are summed from one to three, while Greek subscripts and superscripts are summed from one to four. The space-time metric $(ds)^2 = dx^\mu dx^\mu - c^2(dt)^2$ becomes $(ds)^2 = dx^\mu dx^\mu g_{\mu\nu}$, where the non-zero elements of the metric tensor $g_{\mu\nu}$ are $g_{ij} = \delta_{ij}$ and $g_{44} = -1$. The metric tensors $g_{\mu\nu}$ and $g^{\mu\nu}$, which is defined so that $g_{\mu\nu} g^{\nu\lambda} = \delta_\mu^\lambda$, may be used to raise and lower indices covariantly. The usual three velocity v^i may be extended to a relativistic covariant four-velocity u^μ by the definitions $\gamma^{-2} = 1 - v^i v^i / c^2$ and $u^\mu = \gamma v^i, u^4 = \gamma c$ so that $u^\mu u_\mu = -c^2$.

The electromagnetic field tensor $F_{\mu\nu}$ is antisymmetric and is given by

$$E_i = cF_{i4} = -cF_{4i}, B_1 = F_{32}, B_2 = F_{31}, B_3 = F_{12} = -F_{13}, B_4 = F_{12} = -F_{13};$$

while the Lorentz force on a particle of charge q is $q(\vec{E} + \vec{v} \times \vec{B})$, $= qF^{\mu\nu}u_\nu/\gamma$. The general form of the external magnetic field we are interested can be expressed as

$$B_1 = B_{10} + B_{11}x^1 + B_{12}x^2 + B_{111}x^1x^1 + B_{112}x^1x^2 + B_{122}x^2x^2,$$

$$B_2 = B_{20} + B_{21}x^1 + B_{22}x^2 + B_{211}x^1x^1 + B_{212}x^1x^2 + B_{222}x^2x^2,$$

$$B_3 = B_{30} + B_{31}x^1 + B_{32}x^2 + B_{311}x^1x^1 + B_{312}x^1x^2 + B_{322}x^2x^2,$$

where all the coefficients $B_{10}, B_{20}, B_{30}, B_{11}, \dots, B_{222}$ are functions of x^3 , with B_{10} the dipole, B_{ij} the quadrupole, and B_{ijkl} the sextupole components. The beam distribution function, $f(x^\mu, u)$,

satisfies the relativistic Vlasov equation;

$$\left[\frac{\partial}{\partial t} + \vec{v} \cdot \vec{\nabla} + \frac{q}{m} (\vec{E} + \vec{v} \times \vec{B}) \cdot \frac{\partial}{\partial \vec{u}} \right] f = 0, \quad (1)$$

where m is the particle mass. In a covariant formulation, we can multiply Eq. 1 by γ and rewrite the relativistic Vlasov equation as

$$\left(u^\mu \frac{\partial}{\partial x^\mu} + \frac{q}{m} F^{\mu\nu} u_\nu \frac{\partial}{\partial u^\mu} \right) f = 0. \quad (2)$$

The volume element $d\omega = du^1 du^2 du^3 / \gamma$ in the four-momentum space is invariant under a Lorentz transformation. Since the transverse coordinates, x^1 and x^2 , are invariant under a Lorentz transformation, we define an invariant phase space volume element under a Lorentz transformation to be: $d\Omega = dx^1 dx^2 du^1 du^2 du^3 / \gamma$, and a phase space average

$$\langle X \rangle = h^{-1} \int X f d\Omega, \quad (3)$$

with $h = \int f d\Omega$. The lowest moment of the Vlasov equation (Eq. 2) gives

$$\frac{\partial}{\partial x^3} h(u^1) + \frac{\partial}{\partial x^4} h(u^4) = 0. \quad (4)$$

With Eq. 2 multiplied by u^μ and $u^\mu u^\mu$ then integrated over $d\Omega$, we have

$$\frac{\partial}{\partial x^3} h(u^1 u^\mu) + \frac{\partial}{\partial x^4} h(u^4 u^\mu) = \frac{q}{m} h(F^{\mu\nu} u_\nu) \quad \text{and} \quad (5)$$

$$\frac{\partial}{\partial x^3} h(u^1 u^\mu u^\mu) + \frac{\partial}{\partial x^4} h(u^4 u^\mu u^\mu) = \frac{q}{m} h((F^{\mu\nu} u_\nu u^\mu) + (F^{\lambda\mu} u_\mu u^\lambda)), \quad (6)$$

respectively. There are four independent equations represented in Eqs. 5 and ten in Eq. 6. Equations 4 to 6 are basically the same as the fluid equations of Newcomb⁴ and Amendt and Weitzner⁵, with the additional averaging over the transverse coordinates. If $F^{\mu\nu}$ is independent of the transverse coordinates, then Eqs. 4 to 6 can be reduced to a closed system by assuming the third order correlations are negligible, which is the standard approximation used in truncating most fluid equations. Since $F^{\mu\nu}$ depends on the transverse coordinates, Eqs. 5 and 6 cannot be closed without introducing the spatial moment equations:

$$\frac{\partial}{\partial x^3} h(u^1 x^1) + \frac{\partial}{\partial x^4} h(u^4 x^1) = h(u^1); \quad (7)$$

$$\frac{\partial}{\partial x^3} h(u^1 u^\mu x^1) + \frac{\partial}{\partial x^4} h(u^4 u^\mu x^1) = h(u^\mu u^\mu) + \frac{q}{m} h(F^{\mu\nu} u_\nu x^1); \quad \text{and} \quad (8)$$

$$\frac{\partial}{\partial x^3} h(u^1 x^1 x^1) + \frac{\partial}{\partial x^4} h(u^4 x^1 x^1) = h(x^1 u^1) + h(x^1 u^1) \quad \text{for } i, j = 1, 2 \text{ only} \quad (9)$$

We also define the following second order correlation functions,

$$\langle u^{\mu} u^{\lambda} \rangle = h^{-1} \int f(u^{\mu} - (u^{\nu})) (u^{\nu} - (u^{\lambda})) d\Omega, \quad (10)$$

and similar definitions for the third order correlation functions. With these definitions we can rewrite Eqs. 5 to 9, ignoring all third order correlations, as:

$$D(u^{\nu}) = \frac{q}{m} \langle F^{\nu\mu} u_{\mu} \rangle - h^{-1} \left(\frac{\partial}{\partial x^3} h[u^3 u^{\nu}] + \frac{\partial}{\partial x^4} h[u^4 u^{\nu}] \right); \quad (11)$$

$$D(x^{\nu}) = (u^{\nu}) - h^{-1} \left(\frac{\partial}{\partial x^3} h[u^3 x^{\nu}] + \frac{\partial}{\partial x^4} h[u^4 x^{\nu}] \right); \quad (12)$$

$$D[u^{\lambda} u^{\lambda}] = \frac{q}{m} \langle (F^{\nu\mu} u_{\mu} u^{\lambda}) + (F^{\lambda\mu} u_{\mu} u^{\nu}) - (u^{\lambda}) \langle F^{\nu\mu} u_{\mu} \rangle - (u^{\nu}) \langle F^{\lambda\mu} u_{\mu} \rangle \rangle - \langle [u^3 u^{\nu}] \frac{\partial}{\partial x^3} + [u^4 u^{\nu}] \frac{\partial}{\partial x^4} \rangle (u^{\lambda}) - \langle [u^3 u^{\lambda}] \frac{\partial}{\partial x^3} + [u^4 u^{\lambda}] \frac{\partial}{\partial x^4} \rangle (u^{\nu}); \quad (13)$$

$$D[u^{\nu} x^{\lambda}] = \frac{q}{m} \langle (F^{\nu\mu} u_{\mu} x^{\lambda}) - (x^{\lambda}) \langle F^{\nu\mu} u_{\mu} \rangle + [u^{\nu} u^{\lambda}] \rangle - \langle [u^3 u^{\nu}] \frac{\partial}{\partial x^3} + [u^4 u^{\nu}] \frac{\partial}{\partial x^4} \rangle (x^{\lambda}) - \langle [u^3 x^{\nu}] \frac{\partial}{\partial x^3} + [u^4 x^{\nu}] \frac{\partial}{\partial x^4} \rangle (u^{\lambda}); \quad (14)$$

$$\text{and } D[x^{\nu} x^{\lambda}] = [x^{\nu} x^{\lambda}] + [x^{\nu} u^{\lambda}] - \langle [u^3 x^{\nu}] \frac{\partial}{\partial x^3} + [u^4 x^{\nu}] \frac{\partial}{\partial x^4} \rangle (x^{\lambda}) - \langle [u^3 x^{\lambda}] \frac{\partial}{\partial x^3} + [u^4 x^{\lambda}] \frac{\partial}{\partial x^4} \rangle (x^{\nu})$$

where $D = (u^3) \frac{\partial}{\partial x^3} + (u^4) \frac{\partial}{\partial x^4}$ is the relativistic convective derivative.

3. Space Charge Models

A Chernin's Model

A simple model can be introduced at this point to cast Eqs. 11 to 15 to a concrete form if $F^{\nu\mu}$ is expanded to linear terms of x^1 and x^2 . This system of twenty-eight equations is a close set of moment equations for the twenty-eight moments. The desired form of $F^{\nu\mu}$ is obtained by expanding the magnetic field to linear terms of x^1, x^2 and assuming that Chernin's space charge model¹ for the electric field, which is given by

$$\begin{aligned} E^1 &= \rho(q_{11}(x^1 - (x^1)) + q_{12}(x^2 \\ E^2 &= \rho(q_{12}(x^1 - (x^1)) + q_{22}(x^2 - (x^2))), \\ E^3 &= 0, \end{aligned} \quad (16)$$

where ρ is the line charge density of the beam, $q_{11} = S_2/D$, $q_{12} = S_1/D$, $q_{22} = -[x^1 x^2]/D$, $D = S_0(S_1 + S_2)$, $S_1 = [x^1 x^1] + S_0$, $S_2 = [x^2 x^2] + S_0$, and $S_0 = ([x^1 x^1][x^2 x^2] - [x^1 x^2]^2)^{1/2}$. With this model we can evaluate the moments that involve $F^{\nu\mu}$ in Eqs. 13 to 15.

Equations 13 to 15 can be related to the second order moment equations of Chernin¹ if we restrict them to the same external magnetic field as in Ref. 1, i.e. $B_{10} = B_{20} = 0$ and no sextupole

components. We can identify our notation $[x^{\nu} u^{\mu}]$ with $\Sigma_{\nu\mu}$ in Ref. 1 by the following rules $[u^{\nu} u^{\lambda}] = \Sigma_{2,2}$, $[u^{\nu} x^{\lambda}] = \Sigma_{2,1,2,1}$, $[x^{\nu} x^{\lambda}] = \Sigma_{2,1,2,1} - 1$. Since $[u^{\nu} u^{\lambda}] = [u^{\lambda} u^{\nu}] = [u^{\nu} x^{\lambda}] = 0$ (for $\nu = 1, 2$) for a mono-energetic beam with a delta function distribution in u^3 , Eqs. 13 to 15 is the same set of equations as in Ref. 1, with D equivalent to the ordinary time derivative. Notice that in such a case, the ten second moments (correlations) in the transverse directions form a closed system and are no longer coupled to the zeroth, first and other second moments

B Cylindrical Model with Image Charges

Chernin's model does not take into account the effects of the image charges of the metallic boundary nor the longitudinal component of the space charge fields. To construct an improved space charge model we assume that the charge density, ρ , is given by a collection of charge units;

$$\rho = \sum_{i=1}^N a_i(x_3, t) g(\vec{x} - \vec{z}_i), \quad (17)$$

where g is the distribution of finite size charge elements, e.g. truncated Gaussians, and the location of the charge elements is assumed to be independent of x_3 . Note that g depends on x_1, x_2 and x_3 , while a_i depends only on x_3 . The Green's function inside a metallic cylinder, in the beam frame is

$$G(\vec{x}, \vec{x}') = \frac{4\pi}{a} \sum_{m,n} \frac{J_m(x_{mn} \frac{r}{a}) J_n(x_{mn} \frac{r'}{a})}{x_{mn} [J'_m(x_{mn} \frac{r}{a})]^2} \cos m(\theta - \theta') \cdot \exp(-|x_3 - x'_3| \frac{x_{mn}}{a}),$$

where r and θ are the cylindrical counterpart of x_1 and x_2 , J_m is the Bessel function, x_{mn} is the n th zero of the Bessel function J_m and a is the radius of the cylindrical tube. The electrostatic potential in the beam frame becomes

$$\phi(\vec{x}) = \int G(\vec{x}, \vec{x}') \rho(\vec{x}') d^3 x', \quad (18)$$

assuming the velocity spread in x_3 is not important. After some algebra, the space charge contribution to $\langle F^{\nu\mu} u_{\mu} \rangle$ can be written as

$$\langle F^{\nu\mu} u_{\mu} \rangle = h^{-1} \sum_{i,j} a_i(x_3) \int a_j(x'_3) A_{\nu}(x'_3) d^3 x',$$

where $A_{\nu}(x'_3)$ are the independent precomputed integrals of $g(\vec{x}' - \vec{z}_i)$ and the Green's function. It is interesting to note that space charge effects appear as bilinear terms here compared with the more nonlinear behavior in the Chernin model.

• Work supported by DARPA/DSO

References

- 1 D. Chernin, *Particle Accelerators*, **24**, 29 (1988).
- 2 P. J. Channell, *IEEE Trans. Nucl. Sci.*, **30**, 2007 (1983).
- 3 P. J. Channell, L. M. Healy, and W. P. Lyssenko, *IEEE Trans. Nucl. Sci.*, **32**, 2565 (1985).
- 4 W. A. Newcomb, *Phys. Fluids*, **25**, 846 (1982).
- 5 P. Amendt and H. Weitzner, *Phys. Fluids*, **28**, 949 (1985).

CQL(3D): A Fokker-Planck Transport code for RF and NBI Tokamak Simulations

M.G. McCoy and G.D. Kerbel
National Energy Research Supercomputer Center
and
R.W. Harvey, General Atomics

In this abstract, we will present a general review of the options built into the bounce-averaged Fokker-Planck code, CQL(3D), and of some of the modes in which it can operate. From its beginnings as a non-relativistic 2D (v, θ) kinetic code, it has evolved into a multi-species tokamak code capable of modeling most heating and current drive scenarios, ion-ion and ion-electron collisions, with sufficient dimensionality to permit the code's use as a diagnostic tool for a number of experiments. This 3D, bounce-averaged code is capable of exploring non-Maxwellian features in tokamaks induced by dc electric fields, alpha particles, neutral beams (NB), RF heating, and radial transport, all of which are separate physics modules in the code. The full 2D (v, θ) momentum space distribution of electrons and/or ions is determined on a radial array, ρ , of 2D cross-section plasma flux surfaces, given the steady-state, bounce-averaged distributions $f(v, \theta, \rho, t)$. The bounce-average process, strictly applicable for nearly collisionless plasmas characteristic of many of the most advanced tokamaks, takes into consideration the poloidal angle variation of the distributions, effectively accounting for the 4D nature of tokamak phase space.

The underlying philosophy in the CQL(3D) development has been to apply available knowledge of the physics of tokamak distributions with as much detail and integration as possible, in order to verify that these physics models are truly adequate. "No-free-parameter" modeling of tokamak discharges is the goal; clearly, in the domain of radial diffusion coefficients, this remains a more distant goal, whereas in the domain of velocity space RF diffusion coefficients, the goal is much closer. After describing the time advancement techniques, we will briefly touch upon the major physics packages resident within or called by the code.

TIME ADVANCEMENT

Two options exist for advancing the distribution function(s) on a particular flux surface. For time dependent calculations, or for calculations in which the evolution of the distribution function is of interest, we employ operator splitting. For calculations in which the steady state alone is of interest (by far the most common) we discretize fully implicitly in (v, θ) and then utilize an in-house developed Gaussian elimination solver to solve the system. The solver (tailored for the Cray-2) will solve for 10,000 unknowns (a 100 by 100 discretization) in about .5 sec. When running in this mode, the time step is non-physically large (on the order of several seconds), and boundary conditions are chosen appropriately to conserve particle density. In this case we speak of iterations, not time steps. The procedure utilized to advance the distribution(s) when transport is enabled is described below, in the section devoted to radial transport.

MAGNETIC FIELD GEOMETRY

Assuming that a standard format "endisk" MHD equilibrium file is available, the code reads in the relevant poloidal flux function data and proceeds to determine the flux surfaces to which the particle orbits are bound in this zero banana width approximation. Alternatively, the code has some capability of calling an MHD code using appropriate averages of computed currents and pressures in Ampere's law to produce estimates for p' and $f(p)$ (pressure, $f(RB)$) as input to the MHD code. In this manner, the codes iterate to convergence.

COLLISIONS

The code generally assumes an arbitrary number of time advanced species and background (fixed, Maxwellian) species. The collision operator for each time advanced species is summed over the contributions of the background and (if desired) the time advanced species. This model utilizes the operator as described by Bernstein and Baxter (1), except that the relativistic generalization of the dyadic has been approximated by a Taylor expansion and the error made vanishes with $v \rightarrow u/c^2$. Here the prime represents the field distribution. In cases where the field (or background) species can be considered nonrelativistic, this approximation is adequate. For those cases in which the background species is relativistic, the error can be as large as the corrections Klimontovich has pointed out that for isotropic distributions, the coefficient of the order epsilon vanishes; however, since we are interested in anisotropic field distributions, the error we retain will be of order ϵ in the general case, while in some specific applications (cf. current drive efficiency on Maxwellian field distributions), this represents a significant underestimation of the accuracy of the calculation. Braams and Karney (2) have developed a differential formulation of the Landau-Fokker-Planck collision integral which should be implemented in favor of the current less accurate treatment.

NEUTRAL BEAM INJECTION

In this case, the source of fast ions is derived from the NFREYA (3) code; current drive is obtained from the distorted electron 1d ion distributions. Since the electrons produce a compensating current, this represents an interesting example where the code could usefully time advance two species, ions and electrons, in tandem. Clearly the mesh, if common to both species, must take into account the disparity in velocities of typical electrons and ions, and this is accomplished by packing a large number of the mesh points near $v=0$, from the perspective of the electrons. Analytic estimates exist for the extent of electron compensation, and are in close agreement with the code results. The code iterates to convergence in tandem with the SELENE (4) MHD code.

RF CURRENT DRIVE

We have developed a bounce-averaged quasilinear module applicable to electron Landau/TMP and cyclotron quasilinear diffusion and damping. Minor modifications to produce an ion cyclotron damping and diffusion module are yet to be done. Wave characteristics (RF electric fields and polarizations, trajectories, wave numbers and energy density relations) based on bulk plasma properties, are supplied by an external wave code called by CQL(3D). In this manner, CQL(3D) is presently coupled to the Brambilla (5) lower hybrid/fast wave ray tracing code and the TORCH (6) electron cyclotron ray tracing code. Full consistency of the quasilinearly diffused distribution functions and the associated damping of the rays, is obtained by iteration. This is usually achieved to an error of less than 1% everywhere in p less than 25 iterations. Usually 15 iterations is adequate; detailed calculation of the electron distributions has been used to predict hard x-ray Bremsstrahlung spectra, and the HORACE (7) post-processor has calculated cyclotron emission spectra from the nonthermal distributions. Applications have been made to experiments on the D3D, ASDEX, T-10, and FT-1 tokamaks, and to the ITER conceptual design.

The most comprehensive, and very successful, simulations have been of the lower hybrid current drive experiment on ASDEX. Using the experimental bulk plasma profiles of plasma densities and temperatures, and characteristics of the launched spectra calculated with the Brambilla LH antenna code, it has been shown that combined low and high parallel spectra give marked current profile broadening, in good quantitative agreement with the experiment. The predicted hard x-ray was in broad agreement with the experiment. From strictly central current drive experiments in the same device, there was evidence of fast particle diffusion effects. Similar evidence has arisen in the D3D inside launch EC current drive experiment.

TRANSPORT

Velocity dependent, radial transport terms, describing both diffusion and convection of the plasma species, have recently been added to CQL(3D). Provisions are made to treat the radial diffusion as occurring either at constant equatorial plane velocity coordinates, or at constant energy and magnetic moment. The spatial diffusion and convection terms are *ad hoc*, but can be determined from more detailed treatment such as neoclassical theory, or from separate calculations, for example, with gyrokinetic codes. This approach is similar to the treatment in the BANDIT (8) code. We emphasize here that the full diffusion tensor is not included since the cross derivative terms involving p are ignored. The major reason for ignoring these terms has been computational expediency and inadequate estimates for the size of these coefficients. Without these terms, time advancement is considerably simplified. We use operator splitting through an implicit treatment of radial diffusion and convection terms. First the 2D split is done in momentum space at each flux surface, and then a set of 1D equations is solved in the p direction. The time step is chosen appropriate to dictates of radial diffusive time scales; thus, the system marches to steady state using time steps much larger than would be possible were the full 3D system to be advanced using explicit differencing, and much more cheaply than would be possible were the entire system to be differenced implicitly with the subsequent requirement to solve a system of 20000 unknowns. Possible approaches to avoiding either of these unattractive options will be presented. It is currently of interest to include the full tensor in the code since some theoretical estimates have recently been developed to compute the entire matrix of Fokker-Planck coefficients arising out of wave-particle interactions (9). It is uncertain how this process competes with anomalous diffusion, and the code could be used to examine these questions.

The calculations utilizing transport are compute intensive, involving on the order of 1-4 hours. The lower estimate applies to coarse discretizations of ($v=70$, $\theta=70$, $p=12$), and resolutions on the order of (150,150,20) would be desirable. We are interested in creating a Fokker-Planck transport tool, or "slide rule" which would allow experimentalists to obtain solutions in a few minutes, not hours. This is becoming possible now since 100 gigaflop concurrent devices are promised in the near future. Nonetheless, the procedure will not be easy, since we wish to retain large time steps, and consequently do not plan to utilize explicit differencing and domain decomposition. We will outline some possible approaches to this problem.

REFERENCES

- (1) I.B. Bernstein and D.C. Baxter, *Phys. Fluids*, **24**, p108 (1981).
- (2) B.J. Braams and C.F. Kamey, *Phys. Rev. Lett.*, **59**, no.16, p1817 (1987).
- (3) R.H. Fowler, J.A. Holmes, J.A. Rome, ORNL/TM-6845.
- (4) M. Azumi et al., in *Computing Methods in Applied Sciences and Engineering*, R. Glowinski and J.L. Lions, Eds, (North Holland, 1980) p. 335
- (5) M. Brambilla, *Comput. Phys. Rep.* **4** (1986) 71.
- (6) R. C. Myer, M. Porkolab, G. R. Smith, A. H. Kriz, *Nucl. Fus.* Vol 29, p. 2155 (1989).
- (7) R.W. Harvey, M.R. O'Brien et al., GA-A19643, submitted to *Phys. Fluids*.
- (8) J. S. McKenzie, M. R. O'Brien and M. Cox, to appear in *Phys. Comp. Comm.*
- (9) K. Kupfer, R.W. Harvey and M.G. McCoy, "Fokker-Planck Coefficients for RF Current Drive, Including Wave Driven Radial Transport", IAEA Technical Committee Meeting on Fast Wave Current Drive in Reactor Scale Tokamaks, Arlos (FRANCE) Sept 23-35, 1991.

Thursday Morning
September 5, 1991

Invited Oral Session (OT)
8:30am - 12:30pm

Regatta Ballrooms A & B

C. Birdsall, presiding

A 2D Electromagnetic PIC Code for Distributed Memory Parallel Computers

P. C. Liewer and R. D. Ferraro,

Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109

T. Krucken, California Institute of Technology, Pasadena, CA 91125

V. A. Decyk and J. M. Dawson, Physics Department, University of California, Los Angeles, CA 90024

The two dimensional electrostatic plasma particle in cell (PIC) code described in [1] has been upgraded to a 2D electromagnetic PIC code running on the Intel Delta Touchstone, Intel iPSC/860 and Caltech/JPL Mark III/parallel MIMD computers. Parallelization is achieved through domain decomposition in real and Fourier space. Results from a simulation showing a two dimensional Alfvén wave filamentation instability are shown, these are the first simulations of this 2D Alfvén wave decay process.

Introduction

A standard two-dimensional electromagnetic particle-in-cell (PIC) simulation code has been implemented on MIMD parallel computers. To obtain efficient operation on a distributed memory parallel computer the physical domain is divided in one dimension in sub-domains equal in number to the number of processors so that each sub domain contains the same number of grid points and roughly the same number of particles [3]. The code runs on the Intel Delta, Intel iPSC/860 and the Caltech/JPL Mark III/parallel distributed memory computers.

The code can be used to study wave-particle interactions and instabilities in the linear and nonlinear phase in plasmas. In the first applications we simulated the decay of large amplitude Alfvén waves and compared our results with the decay instabilities predicted by theoretical models based on linearized two fluid MIMD models [5,6]. Large amplitude Alfvén waves are observed in the solar wind, upstream of the Jovian bow shock and near other interplanetary shocks. Understanding the decay of these waves will help understand spacecraft observations and increase our understanding of low frequency plasma magnetic turbulence.

Code Description

The code solves the equation of motion for the plasma particles and the two Maxwell's equations for the electromagnetic field [2]. A particle with velocity \vec{v}_p at position \vec{x}_p is subject to the Lorentz force

$$\frac{d\vec{v}_p}{dt} = \frac{q_p}{m_p} \left(\vec{E}(\vec{x}_p) + \frac{\vec{v}_p}{c} \times (\vec{B}(\vec{x}_p) + \vec{E}_0) \right).$$

$$\frac{d\vec{x}_p}{dt} = \vec{v}_p$$

\vec{E}_0 is a constant uniform external magnetic field. \vec{E} and \vec{B} describe the self-consistent electromagnetic field which is a solution of Maxwell's equations. Variations are allowed in two directions (\vec{x}_p and \vec{y}_p), but all three components of \vec{v} , \vec{E} and \vec{B} are needed. A spectral method is used to solve the field equations on a discrete grid with periodic boundary conditions (because of the explicit time dependence of the equations it is very difficult to implement other boundary conditions). In Fourier space Maxwell's equations can be split in a transverse and a longitudinal part:

$$\frac{1}{c} \frac{d\vec{B}_\perp(\vec{k}, t)}{dt} = -i\vec{k} \times \vec{E}_\perp(\vec{k}, t)$$

$$\frac{d\vec{E}_\parallel(\vec{k}, t)}{dt} = i\vec{k} \times \vec{B}_\perp(\vec{k}, t) + 4\pi \vec{k} \cdot \vec{J}_\parallel(\vec{k}, t)$$

1

$$\vec{E}_{\text{ext}}(\vec{k}, t) = -4\pi \frac{i\vec{k}}{k^2} \rho(\vec{k}, t)$$

where

$$\vec{A}(\vec{k}, t) = \sum_{\vec{r}, \vec{r}'} \vec{A}(\vec{k}, \vec{r}, t) e^{i(\vec{k} \cdot \vec{r} + \vec{r} \cdot \vec{r}')}.$$

$$\vec{k} = k_x \vec{e}_x + k_y \vec{e}_y.$$

The transverse and longitudinal components of a mode of \vec{A} are defined as

$$\vec{A}_{\text{trans}}(\vec{k}, t) = \frac{\vec{k}(\vec{k} \cdot \vec{A}(\vec{k}, t))}{k^2}.$$

$$\vec{A}_{\text{long}}(\vec{k}, t) = \vec{A}(\vec{k}, t) - \vec{A}_{\text{trans}}(\vec{k}, t)$$

so that $\vec{k} \cdot \vec{A}_{\text{long}}(\vec{k}, t) = \vec{k} \cdot \vec{A}(\vec{k}, t) = 0$. The source terms in Maxwell's equations, $\rho = \sum_p q_p n_p$ and $\vec{J} = \sum_p q_p \vec{v}_p n_p$, are the charge and current density respectively due to the plasma particles, with n the simulation volume.

Each time step in the PIC code proceeds in two stages. In the first stage, the particle equations are solved. The fields at the position of the particle is found by interpolation from the fields at the grid points. The new charge and current densities are found by interpolation of the particle information to the grid points. A quadratic interpolation scheme is used. In the second stage, the charge and current densities are Fourier-transformed and the field equations solved using an explicit leapfrog algorithm. The new fields are transformed back to real space in preparation for the next set of particle updates.

Runtime diagnostics include evaluation of particle distribution functions, particle energy and momentum flux, drift velocities, overall momentum and energy, various plots of the electromagnetic fields as well as evaluation of the field momentum and energy. These diagnostics, including the plots, are all performed in parallel.

Parallel Aspects of the Code

The code runs SPMD (single program multiple data -all node programs are identical) style under Fortran Cubix on the Mark III/parallel and under Express Fortran Cubix on the Intel Delta and Gamma machines. Parallelization is achieved by means of one dimensional domain decomposition in real and Fourier space. Each processor knows the particles and grid quantities in his subdomain in real space (x, y), which is a horizontal strip in Fourier space (k_x, k_y). The subdomains are vertical strips. These decompositions are described in Ref. 1.

Interprocessor communication is necessary at several stages. The 2D FFT is done as two sets of 1D FFTs, first in the horizontal, then in the vertical direction. Between these two operations the entries $f(k_x, y)$ are rearranged among processors, such that each set of 1D FFTs can be done in parallel. This requires considerable interprocessor communication. Information about particles moving into a neighboring processor's subdomain must also be passed to the new processor's particle list and removed from the old processor's list. Additional communication is also necessary for the diagnostics.

Most of the computational time is needed to advance the particles. Due to the nonuniformity of the magnetic field the particle push is in an electromagnetic code much more time consuming than in an electrostatic PIC code. To obtain efficient operation it is therefore very important that the number of particles in each processor is roughly the same. In the first applications we have seen no major density fluctuations so that dynamic load balancing [4], i.e., changing subdomain boundaries during the run, was not necessary. The code has been run on the Caltech/JPL Mark III/parallel (up to 128 processors) with the Weick XL processor and the Intel iPSC/860 (up to 64 processors). A fixed problem on the same number of nodes runs roughly three times faster on the Intel iPSC/860 than on the Mark III/parallel. For a problem with 204,312 particles and a 256x256 grid run for 150 time steps on 32 processors, the Intel total time was 373 secs versus a Mark III/parallel time of 1043 secs. Times for runs on the Intel Delta "Touchstone" computer will also be presented.

2

Application to 2D Decay of Large Amplitude Alfvén Waves

The code can be used to study wave phenomena, wave-particle interactions, kinetic effects, and instabilities in the linear and nonlinear regimes in rectangular 2D geometry. In the first applications we have used it to simulate the behaviour of large amplitude circularly polarized parallel (to the static magnetic field) propagating Alfvén waves as observed in the solar wind and other interplanetary environments. These waves are subject to parametric decays into other plasma waves caused by nonlinear terms. The initial "pump" Alfvén wave (ω_0, k_0), $E_0 \parallel \hat{B}_0$, beats with a daughter wave (ω, k) which is initially excited from the noise and forms a pair of sideband waves (ω_{\pm}, k_{\pm}). Frequencies and wavenumber matching requires that $\omega_{\pm} \approx \omega_0 \pm \omega, k_{\pm} \approx k_0 \pm k$. If all waves are normal modes (eigenmodes), i.e., all pairs ω, k are solutions of the plasma dispersion relation, the instability has a large growth rate. For high amplitude pump waves it is, however, often sufficient that three waves have eigenmodes.

Parametric instabilities of Alfvén waves have been discussed in numerous theoretical papers based on plasma fluid equations (see for example [5,6]) and the papers cited there). The important processes for our simulations are the $\delta\omega$ instability and the filamentation instability. In the first case, the daughter wave is an ion sound wave with a shorter wavelength than the pump Alfvén wave and the sidebands are electromagnetic waves. The decay products can propagate parallel to the pump wave or at oblique angles; generally the fastest decay occurs for parallel decay products. The one-dimensional decay processes can be studied in simpler 1D models.

On the other hand, the filamentation instability can not be treated in a one-dimensional model since the daughter "wave" is a purely growing, magnetostatic perturbation, i.e., $Re(\omega) \approx 0$, with $E \perp \hat{B}_0$. The pump wave gets "ducted" in density/B-field filaments due to this magnetostatic perturbation. Sidebands are obliquely propagating electromagnetic waves. This wave has only recently been discussed theoretically [6]. Most theoretical study of these decay processes are limited to linearized fluid treatments; particle simulations are used to include kinetic and nonlinear effects on the evolution of these processes. Here, results from the first 2D simulations showing the 2D filamentation instability are presented.

Figure 1 shows the time history of the amplitude of the pump wave in a simulation with a right hand circularly polarized Alfvén wave of amplitude $B/B_0 \approx 0.5$ propagating parallel to an external magnetic field B^0_z . For this run, β (ratio of plasma to magnetic pressure) = 0.3, $m/m_e = 16$, $k_0 v_A/v_{Te} \approx 0.3$, where v_A is the Alfvén speed, and $\omega_{pe}/\omega_{ce} = 1$. The initial pump Alfvén wave is excited by applying an external oscillatory current for a short time at the beginning of the run (Fig. 1). This pump wave is then observed to decay in time, transferring its energy to other modes and to the plasma particles. In the computer simulations, many different decay processes occur simultaneously. The various decay instabilities are separated by using a post processor to analyze the time history and power spectrum of various Fourier modes.

At early times in this simulation, the dominant decay process was a 1D decay instability, with the daughter wave an ion acoustic wave with $k_{\pm}^2 = 1.5k_0^2$ and an Alfvén wave sideband with $k_{\pm}^2 \approx 0.5k_0^2$. Power spectra from the post processor show that these waves also satisfy the frequency matching condition $\omega_0 \approx \omega_{\pm} + \omega$ with $\omega/\omega_{ce} \approx 0.45$, $\omega/\omega_{ci} \approx 0.4$, and $\omega \approx 0.05$. The growth rate of this decay instability is $\gamma/\omega_0 \approx 0.1$, consistent with theoretical predictions for similar parameters [6].

In addition to the 1D decay process, two 2D filamentation decays are observed to grow with lower growth rates than the dominant 1D decay process. Because of the several competing processes, the growth phase observed for these processes are not purely exponential. These processes are identified as filamentation instabilities because, at early times, a perpendicularly propagating daughter wave with zero frequency is observed to grow more or less simultaneously with an obliquely propagating wave with a frequency approximately equal to the pump wave. Both daughter and sideband waves have both density and magnetic perturbations as expected for these electromagnetic modes. For one filamentation instability, the zero frequency daughter wave has $k_{\perp}^2 = 2k_0^2$, the sideband has $k_{\perp}^2 = 2k_0^2$, and the growth rate $\gamma/\omega_0 \approx 0.08$. For the other filamentation instability, the zero frequency daughter wave has $k_{\perp}^2 = k_0^2$, the sideband has $k_{\perp}^2 = k_0^2 + k_0^2$, and the growth rate $\gamma/\omega_0 \approx 0.05$. These growth rates are of comparable magnitude to theoretical predictions [6]. Other waves are also excited in the course of the simulation; even at relatively early times, the situation in these fully nonlinear simulations is far from the ideal situation assumed in the linear analysis [5,7].

As the pump wave decays, the energy goes into exciting other modes. Energy is then transferred to

the electrons and ions via electron and ion Landau and ion cyclotron damping of the obliquely propagating waves. Substantial heating of both ions and electrons is observed. Figure 2 shows the global energy balance for the same case as shown in Figure 1. The field energy (w_e) includes all the energy in the electric and magnetic fields of all modes in the simulation; the ion energy (e_k) is the total kinetic energy of the ions in the simulations and the electron energy (e_e) is the total electron kinetic energy. The large oscillations at early times are due to the exchange of energy between the ions and the fields in the pump Alfvén wave. Note that at later times, the electrons appear to be responsible for most of the observed damping of the field energy. At late times, the density fluctuations associated with the filamentation decays are observed to damp out. Such electron damping can only be studied with full particle-simulation code.

Acknowledgements

One of the authors (T.K.) was supported by the Max-Planck-Institut für Plasmaphysik (IPP) at Garching and the Max-Planck-Gesellschaft at München, Germany. The work was supported in part by the NSF under Cooperative Agreement No. CCR-8809615 and in part by NASA/Heliophysics Physics. A portion of the research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

R.D. Ferraro, P.C. Liewer, V.K. Decyk, "A 2D Electromagnetic PIC Code for the Mark III Hypercube. Proceedings of 5th Distributed Memory Computing Conference, Vol. 1, p. 410 IEEE, 1990.

C.K. Birdall, A.B. Langdon, *Plasma Physics via Computer Simulation*, McGraw-Hill, New York, 1981.

P.C. Liewer, V.K. Decyk, "A General Concurrent Algorithm for Plasma Particle-in-Cell Simulation Codes, J. Comp. Phys. 85, p. 302, 1989.

P.C. Liewer, E.W. Leaver, V.K. Decyk, J.M. Dawson, "Dynamic Load Balancing in a Concurrent PIC Code on the JPL/Caltech Mark III Hypercube, Proceedings of 5th Distributed Memory Computing Conference, Vol. 2, p. 939, IEEE, 1990.

A.F. Viñas, M.L. Goldstein, "Parametric Instabilities of Circularly Polarized, Large Amplitude Dispersive Alfvén Waves: Excitation of Parallel Propagating Electromagnetic Daughter Waves, to be published in Plasma Physics, 1991.

A.F. Viñas, M.L. Goldstein, "Parametric Instabilities of Circularly Polarized Large Amplitude Dispersive Alfvén Waves: Excitation of Obliquely Propagating Daughter and Sideband Waves, to be published in Plasma Physics, 1991.

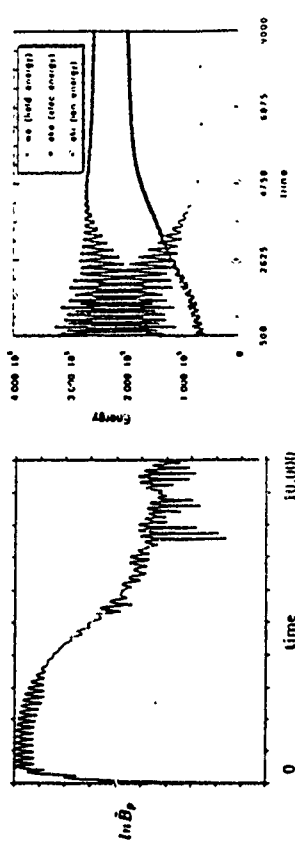


Fig. 1 Time history of the amplitude of the magnetic perturbation of the pump wave

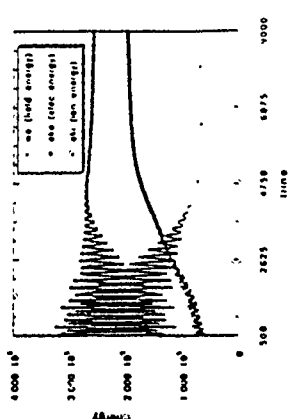


Fig. 2 Time history of simulation energy balance

3-D Electromagnetic PIC Simulation on the NRL Connection Machine*

E. Zaldman

Electronics Science and Technology Division
Naval Research Laboratory
Washington, D.C. 20375-5000

The design of vacuum electronic devices is becoming sufficiently computationally intensive that a 3-D electromagnetic design code has been implemented on the NRL Connection Machine. The code design has been influenced by the requirements of a parallel orientation of coding style and by the complex boundaries typical of currently envisioned devices. The desire to keep the task as simple as possible so as to construct this code within a useful time frame was an additional influence.

Simulation Model

The simple aspects of the task arise due to the success of electromagnetic PIC in the simulation of RF generation in vacuum electronics. This is because the fundamental Maxwell and Newton-Lorentz equations are used to calculate the particle-field interactions self-consistently. The principle bottleneck for sequential computer architectures (including vector architectures) is the massive number of particles required for adequate realism of the simulations. To determine why this is so and how to get around this problem, the principle algorithmic areas in a PIC simulation code should be examined. It is useful in the current context to divide the PIC algorithm into four principle components: 1) conversion of particle quantities such as charge and current into quantities defined on a computational grid; 2) solution of the field quantities at these grid positions; 3) interpolation of these fields back to the particle locations; so that 4) the particles are advanced forward in time.

* Work supported by the Office of Naval Technology and the Office of Naval Research

The model is then further complicated by the intended application. The interaction of an electron beam with the electromagnetic fields in a structurally bounded region of space for the purpose of RF generation indicates a complex boundary problem. Conducting walls may be of unusual geometries, beam emission may be dependent upon local spatial quantities, and highly disparate spatial scales may be required to adequately describe the fine scale physics in one region of the device while the overall device physics may also be required.

Parallel Aspects of the Implementation

The way in which the code is mapped onto the computer is determined by the number of CPU's and how their connectivity is configured. For a massive number of CPU's, the particles may each have their own CPU as may the field grid nodes. With each particle having its own CPU, the limiting factor is how to interpolate (communicate) to/from the field data (at the field CPU's). If the particle CPU is identical with the field CPU, the communication time is essentially nil. A partition or bin sort may be used to achieve this, with the overhead up front in the sorting. If the particle CPU is near the field CPU in the sense of communication links, savings could still be achieved. The current version of the code uses a linear ordering of particles with no sorting, but with a cell pointer. The particle processor set is thus arranged linearly. The particle push itself is completely parallel.

The field grid processor set connectivity is logically arranged as a regular 3-D grid. The physical grid, however, may be non-uniform. The finite difference local formulation of the field solver lets the communication between CPU's be of the nearest neighbor form. The nearest neighbor communication pattern is Gray encoded upon the fundamental hypercube communication pattern in an optimized manner.

The interaction between the particle processor set and the field processor set (actually the same physical processors) is where the bottlenecks appear. Scattering the field information out to the particles

is relatively straightforward with the use of a cell indexing pointer. The particle CPU asks for information from the field CPU's as determined by this pointer. This pointer is updated as the particle moves from cell to cell. This is currently done by boundary comparison, however the use of the position weighting factors is also possible (and probably better). Gathering the particle information to the field locations is done from the viewpoint of the particles. The particles send their data to each of the eight grid CPU's defining their bounding cell's corners. Linear weighting is used for both communications.

Boundary conditions are also applied in parallel. Conducting walls, absorbing layers and particle emission surfaces are currently operational. The diagnostics are implemented with parallel forms. The graphics processing is done primarily in parallel, with just the resultant points or lines passed to the front end system and annotated with labels, etc.

Application and Timings

The driving force behind this work has been the development of a code useful for the development of vacuum electronic devices for RF generation and amplification. A comparison with codes on other machines is not completely straightforward as no direct port of this code exists. Individual components may be compared, and items such as the time per particle per time step can be broken out.

As this is a brand new code, only a portion of its capabilities are stressed at this time. The first application, the gyro-bwo, is a good test of the boundary physics conditions. In this simulation, the 3-D aspects, and the physics oriented boundary conditions are most important. The grid is, however, simple. Some diagnostic results will be shown.

Future Directions

Now that a working code has been developed, the next steps are to extend its usefulness. A conversion to CMFortran is planned for the near future. The integration of a friendly, X-Window based user interface is also being designed. The overall design tool thus obtained is expected to be heavily used in future vacuum electronics design in the near future.

14th International Conference on the Numerical Simulation of Plasmas

Plasma PIC Simulations on MIMD Computers

R. D. Ferraro and P. C. Liaw

Jet Propulsion Laboratory, California Institute of Technology

and

V. K. Decyk

Department of Physics

University of California, Los Angeles

In previous work we have demonstrated the efficiency of 1D and 2D electrostatic and electromagnetic PIC codes on the JPL/Caltech Mark III/4 Hypercube 1-3 using the General Concurrent PIC (GCPIC) algorithm¹. We have also ported our codes to the Intel 1860 Gamma machine and have been selected as an early user of the Intel Delta machine at the California Institute of Technology. The GCPIC algorithm is a general approach to parallelizing PIC codes which is applicable to many different parallel architectures. We have recently extended our work to include dynamic load balancing of particles⁴ to improve the efficiency of code execution when density inhomogeneities develop in the course of a simulation. In this paper, we discuss extending our work to 3D simulations and the necessary improvements required for full use of the Intel Delta machine.

The Mark III/4 Hypercube and the Intel 1860 Gamma machine have 2d processors ($0 < d < 8$), each with some amount of local memory (The Mark III/4 has 4 Mb per processor, the Intel up to 16 Mb). The processors are connected via communication channels in a hypercube topology, i.e., the connections between processors may be mapped onto the edges of a d -dimensional hypercube. The Intel Delta machine has a mesh communication topology with processors, I/O servers, and user service processors arranged in a 16 by 32 rectangle. Since each processor has direct access only to its own local memory, the particles and grid cells in a PIC simulation must be distributed in some fashion among the available processors. Since the particle push represents the major fraction of the computation time, it is essential to have approximately equal numbers of particles in each processor. The field grid must be distributed as well for the purpose of solving for the new fields, and in a manner which is not necessarily the same as that needed to push the particles.

The essential elements of the PIC code, the particle push and the field solver, are parallelized in GCPIC by partitioning the field grid and particles among the processors of MIMD (Multiple Instruction - Multiple Data) computers in two separate data decompositions. In the

primary decomposition, particles and field grids are assigned to processors so that the time advance of particles and the charge deposition may be done efficiently in parallel without the need to communicate with other processors. In the secondary decomposition, the charge and field grids are assigned to processors so that the field solver may execute efficiently. At each timestep, field quantities must be transformed between these two partitions, and some particles must be transferred among processors as they traverse the simulation space.

In our 2D load balanced code, we use 1D partitions (Fig. 1) for the primary decomposition. The size of these partitions are adjusted during the simulation so that each processor has approximately the same number of particles. The partition sizes are determined by calculating a 1D number density function along the partition direction from the deposited charge density. The partition boundaries for each processor are determined by requiring

$$\frac{N_{\text{particles}}}{N_{\text{processors}}} = \int_0^{y_1} dy \int dx n(x,y)$$

$$(p+1) \frac{N_{\text{particles}}}{N_{\text{processors}}} = \int_0^{y_1} dy \int dx n(x,y)$$

where $\int dx n(x,y)$ is the 1D density function calculated from the deposited charge, p is the processor number, and y_1 and y_2 are the left and right partition boundaries. Particles are migrated to the processor whose boundaries encompass its position. Between repartitioning timesteps, some particles will cross their assigned processor's boundaries as they are time advanced. These must be passed on to the processor responsible for their new position.

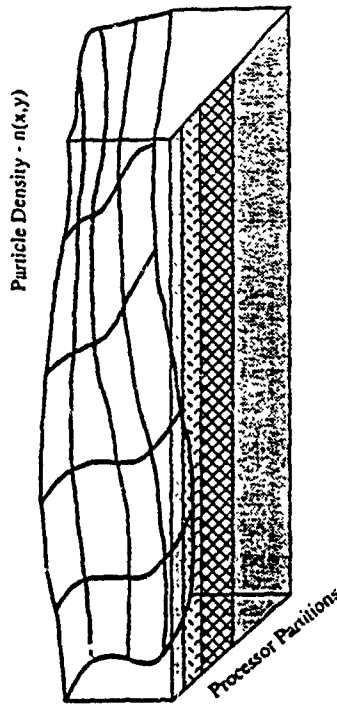


Figure 1. An example of irregular primary decomposition partition boundaries. The size of the partitions are adjusted during the course of the simulation so that all processors have approximately the same number of particles.

The field solver used in our codes is FFT based. For best parallel efficiency, the secondary decomposition should be a regular partitioning of the field grid among processors. When the primary decomposition is also regular, transforming from one partition to the other involves only the exchange of guard cell information. This can be accomplished using nearest neighbor communication. For an irregular primary decomposition, a more general redistribution of grid based information is required (Fig. 2). This involves substantially more overhead than the nearest neighbor case. The FFTs do not parallelize as efficiently as does the particle push, making the solver a significant fraction of the timestep loop. If the total particle to grid ratio is not sufficiently high, the extra overhead associated with irregular partitions can nullify the effects of dynamic load balancing. In this situation, the code actually runs slower than it would with fixed regular partitions for the particles.

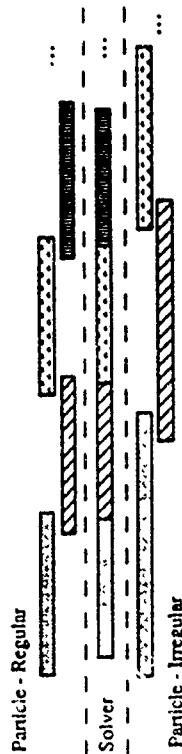


Figure 2. An example of field grid assignments to processors for regular and irregular particle partitions, and a regular solver partition. Some grid information in the particle partitions is duplicated in neighboring processors so that interpolation can be done without accessing neighboring processor memory. In the regular particle partition, only edge information need be communicated to transform to the solver partition. In the irregular particle partition, a general redistribution of grid information is required.

Diagnostics are done in parallel, including graphics, by using one of the field decompositions described above. Phase space plots, for example, are parallelized using the primary decomposition while contour plots of potential are done using the secondary field decomposition. The graphics software operates in parallel, with each processor drawing a separate portion of the graph corresponding to its part of the diagnostic.

For 2D simulations or for the case when the numbers of processors exceeds the number of grids in the smallest dimension, 1D partitioning is not sufficient. Our methods can be readily extended to 2D partitions. However, the number of neighboring processors with which grid information must be exchanged in a regular 2D partition is 8 (4 edges + 4 corners). For regular 3D partitions, the number of neighbors is 26 (6 faces + 12 edges + 8 corners). Irregular partitions are correspondingly worse. The exchange of information required to transform between the primary

and secondary decompositions will impose a substantially higher overhead burden and lead to reduced efficiency. It seems unlikely that extending our approach to 3D will ever be desirable.

To mitigate the overhead of 2D partitions, we are investigating methods of eliminating the transformation between the primary and secondary partitions required at each timestep. This might be accomplished by using a field solver which can be made to operate efficiently in the primary decomposition. The communication overhead would then be reduced to that required for the field solver alone. Potential approaches include finite difference solvers and "out of core" FFT algorithms.

Benchmark timings of our codes have been done on the Mark III Hypercube, the nCUBE2 and the Intel i860 Gamma machines. We anticipate also having timing results for the Intel Delta machine.

1. P.C.Liewer and V.K.Decyk, *J. Comp. Phys.* **85**, 302 (1989)
2. P.C.Liewer, E.W.Leaver, V.K.Decyk, J.M.Dawson, "Dynamic Load Balancing in a Concurrent PIC Code on the JPL/Caltech Mark III Hypercube," in *Proceedings of the 5th Distributed Memory Computing Conference*, IEEE Computer Society Press, 1990
3. R.D.Ferraro, P.C.Liewer, V.K.Decyk, "A 2D Electrostatic PIC Code for the Mark III Hypercube," in *Proceedings of the 5th Distributed Memory Computing Conference*, IEEE Press, 1990.
4. R.D.Ferraro, P.C.Liewer, and V.K.Decyk, "Dynamic Load Balancing for a 2D Concurrent PIC Code," submitted to *Journal of Computational Physics*

The Numerical "Laboratory" for Plasma Simulation: A Modern Approach to the Construction and Maintenance of a Large Simulation Code.

M. J. LeBrun
Japan Atomic Energy Research Institute

M. G. Gray, G. Furnish, and T. Tajima
Institute for Fusion Studies
The University of Texas at Austin

W. D. Nystrom
Los Alamos National Laboratory

Large scale simulation has been very successful as a design and diagnostic tool in a wide variety of applications (integrated circuits, jet aircraft, nuclear reactors, to name a few). Similar application of large codes to gain a predictive capability in plasma confinement may be possible, in light of algorithmic improvements, improved software developing tools, and faster computers. The success of this effort depends on many secondary factors. Foremost is the choice of a numerical algorithm appropriate for the physical situation, with the understanding that the algorithm "best" suited is highly problem-dependent and evolves in time. Of equal or greater importance, however, are the practical issues of coding and maintaining a large and complex simulation code.

Although many of the ideas expressed in this poster are applicable to any large scale simulation project, our central focus is with plasma confinement and transport in a toroidal device such as the tokamak. The simulation of a tokamak shot may well lie outside the range of today's most sophisticated numerical algorithms and most powerful computers. However, we believe the existing numerical models and computer hardware haven't come close to being thoroughly exploited yet.

By utilizing various state of the art algorithms (e.g. general geometry, δf , implicit, gyrokinetic, and subcycling methods) and powerful new computers (from fast workstations like the IBM RS/6000 to massively parallel machines like the Connection Machine), a practical kinetic tokamak simulator may in fact be within reach. Even if the lofty goal of simulating a tokamak shot proves impossible, there are many benefits of this approach. Besides the physical insight gained through more realistic simulation runs, this research provides: (a) a framework in which new algorithms can be evaluated without requiring reinvention of "support" facilities such as mathematical libraries, I/O, graphics, etc., and (b) crucial experience in the construction of such a plasma simulator.

This "Numerical Laboratory" for plasma confinement and transport studies must contend with many obstacles and goals. For one, the system as a whole is very complex. Although perhaps not quite as complex as an actual experiment, it may be orders of magnitude more complex than the typical theory. One can envision a simulation with all of the following effects:

- Instability via many different mechanisms simultaneously (e.g. drift mode, η -mode, ballooning mode).
- Instability for multiple helicities.
- Instability at different radial regions.
- Source and loss term dependence.
- Collisionality dependence.

Diagnosis of the simulation then becomes a large problem in its own right. In addition there are different levels of physics of interest, e.g.

- Time scales: electron vs ion times, etc
- Spatial scales: local vs global
- Different geometrical configurations: slab, cylindrical, toroidal, toroidal with generalized cross section, dipole (e.g. magnetosphere), etc.
- Different boundary and initial conditions
- Interface with other physics, such as radiation, atomic physics, etc.

There are also practical problems associated with moving from a "small" numerical model (single problem/single researcher) to a "large" one (multiple problems/multiple researchers). The choice of programming language is critical, in that it must be flexible and powerful enough to encourage the development of modular, hierarchical, and extensible software. System dependent code must be eliminated so that multiple computing platforms can be supported. The user interface must remain clear and simple yet support a variety of problems and perhaps hundreds of possible input settings. Finally, a large package must be split consistently into smaller pieces (source files, header files, perhaps multiple directories) and maintenance automated (e.g. via the Unix "make" utility).

Many of the elements of the "Numerical Laboratory" are present in an existing code, TPC (Toroidal Particle Code), developed at the IFS [2, 1]. TPC exceeds 60,000 lines and is written in MPPPL, a public domain Fortran preprocessor. MPPPL provides flow control enhancements in the spirit of Fortran 90, resulting in a clear and robust program design. Also it contains a powerful macro language based on the Unix m4 macro processor. Both MPPPL and TPC are highly portable, running on a wide variety of systems including CTSS, VMS, and many variants of Unix (e.g. A/IX, SUNOS, DG/UX, UNICOS).

Some elements of TPC include:

- Modular code construction. Multiple dynamics types (see below), field solvers, switchable via MPPPL macro variables and conditional compilation. User-specific code enhancements handled similarly.
- Toroidal (r, θ, C) type metric. Global ($r = 0$ to $r = a$) or local ($r = r_{min}$ to $r = r_{max}$) radial discretization. General mesh cross section capability under development.
- Two or three-dimensional in field representation. Flexible choice of mesh, including optional nonuniform radial mesh. Grid or mode expansion in azimuthal coordinate. Code can operate in slab or cylindrical limit.

- Electrostatic or Darwin model (see related poster).
- Iterative toroidal field solver. Splits nonseparable Laplacian into invertible and noninvertible parts. Solve invertible part via finite differencing in r , Fourier analysis in θ and ζ . Include noninvertible part iteratively. Convergence scales with inverse aspect ratio, typically less than 10 iterations required.
- Electrons dynamics types (switchable) :
 - Full kinetic, decentered (Lorentz pusher)
 - Drift kinetic. Nuthrup or Littlejohn (phase space preserving) equations
 - Boltzmann response.
 - δf response (Electrostatic only).
 - Optional pitch angle collision operator.
- Ions dynamics types (switchable) :
 - Full kinetic (Lorentz pusher).
 - Drift kinetic (as above).
 - Gyrokinetic (Electrostatic only)
 - δf response (Electrostatic only).
- Post-processor with independent modules for:
 - Spectral analysis and display
 - Time evolution of potential
 - Snapshot diagnostics (density, temperature, etc.)
 - Transport analysis (diffusion, fluxes, as a function of r, t)
 - Test particle trajectories
 - Interferogram analysis

Data stored in file families, allowing data to be off-loaded to cheaper storage media while the simulation runs. Currently native data format is used, though conversion to system-independent format is underway.

- System independent and extensible graphics interface library. We utilize system-independent calls to do specified generic operations. This can be on a high level, i.e. creating an entire plot with one call, or a medium level, requiring several calls to create a single plot. These are implemented via interface routines to the library in question. The choice of graphics library and interface routines is made at compile-time, via the setting of a single macro flag called GRAPHICS. An options-setting package allows unlimited enhancement of graphics routines without requiring change to existing calls. This is accomplished via calls to intermediary routines which set up parameters in advance of the actual plot call (modeled somewhat loosely after the NCAR AUTOGRAPH interface). Also we have developed a library of routines that provides hierarchical multi-windowing (multiple plots per page) via a simple call interface.

- Object and executable file construction controlled by a single master makefile, written in m4. System dependencies (compiler, libraries, etc) are handled by building a makefile for each system (e.g. X m4 -DSUNOS <makefile.m4> >makefile on a Sun).

- Macro-based data structures. Program variables (declarations, common statements) are kept in a single data structure, which must be expanded by routines needing access. These are further subdivided into header files, included only into files needing access. Thus a limited form of data hiding occurs.

- Internal macro-based debugging & error control package. Maintains call stack (printed when an error is detected), allows user to select a trace he made of run from the input file, and other system-independent debugging features.

- and so on ...

In the poster we shall go into further detail on how the various algorithms may be combined. One of the more difficult issues is geometry; however by considering a somewhat general geometrical configuration from the start, we find that new algorithms can be added to TPC relatively easily.

We emphasize that in construction of the "Numerical Laboratory", the computer science issues must not be left to lag. The utilization of a modern programming language, existence of good graphics and visualization tools, source file organization, user interface and internal error control ultimately controls what physics we can get out of such a code.

References

- [1] M. J. LeBrun and T. Tajima. *Particle Simulation in Curvilinear Coordinate Systems*. Technical Report 374, Institute for Fusion Studies, University of Texas at Austin, 1989.
- [2] M. J. LeBrun and T. Tajima. *Particle Simulation of Toroidicity-Induced Drift Modes*. Technical Report 347, Institute for Fusion Studies, University of Texas at Austin, 1988.

3D Electrostatic Gyrokinetic Simulations on Parallel Architectures

J. V. W. Reynders, S. S. Roy, *and W. W. Lee
Princeton Plasma Physics Laboratory

Abstract

A 3D electrostatic gyrokinetic code in sheared slab geometry is presented and performance on a variety of parallel architectures is reported. It is shown that these architectures out perform their serial counterparts in speed of simulation and grid resolution. Parallel architectures and algorithms are discussed and compared.

The 3D electrostatic code reported here uses the gyrokinetic particle simulation model developed by Lee (JCP 72, 213 (1987)). The gyrokinetic Vlasov equation in slab geometry can be written in Klimontovich form as

$$F(x, v_{\parallel}, t) = \sum_{j=1}^N \delta(x - x_j) \delta(v_{\parallel} - v_{\parallel j}) \quad (1)$$

with the phase space trajectories given by

$$\frac{dx}{dt} = v_{\parallel} b_0 + \frac{c}{B_0} E_L(x) \times b_0 \quad (2)$$

and

$$\frac{dv_{\parallel}}{dt} = \left(\frac{q}{m} \right) E(x) b_0 \quad (3)$$

*Present Address: Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544

where $b_0 = \hat{s} + \hat{z}$, and the longitudinal part of the electric field is given by $E^L = -\nabla \phi$. The gyrokinetic Poisson equation is

$$\left(\frac{r_s}{\lambda_D} \right)^2 (1 - \Gamma_0(b)) \phi = -4\pi e (n_i - n_e) \quad (4)$$

where r_s is the ion Larmor radius measured at the electron temperature, λ_D is the Debye length, $b = k_{\perp}^2 r_s^2$, and $\Gamma_0(b) = I_0(b)^2$ where I_0 is the zeroth order modified Bessel function. A pseudo-spectral technique is used to calculate ϕ and a subtracted dipole scheme to interpolate the charged rings to and from the grid upon which the fields are calculated. The charged rings are approximated by an eight-point averaging scheme which resolves modes with $0 < k_{\perp} r_s \lesssim 5$. The trajectories of the rings and the fields that they produce are updated with a predictor corrector method.

The CM-II is comprised of 4096 nodes which each contain 16 single bit processors with a clock speed of 8 MHz. These nodes of 16 processors are paired together with a floating-point interface chip, a 10 MFLOP Weitek 3132 floating-point execution, and 4 Mbytes of memory. Thus, the CM-II has a total of 65536 single bit processors, 2048 Weitek chips, and 8 Gbytes of memory. The processor interconnect scheme uses both hypercube routing and nearest-neighbor communication. The processors are all executing the same set of instructions on different blocks of memory making the CM-II a SIMD (Single Instruction, Multiple Data) machine. There are compilers for Lisp, C, and Fortran each of which have language extensions to utilize the data-parallelism.

The Intel IPSC/860 is a collection of up to 128 very high speed microprocessors each with a clock speed of 40 MHz which communicate through a message based hypercube interconnect. Each processor has 16 Mbytes of RAM for program and data storage and none is shared between processors, making it a MIMD (Multiple Instruction, Multiple Data) machine. There are compilers for FORTRAN and C which support parallelism through calls to message passing routines.

The bottle-neck in particle-simulations on parallel architectures is the memory bandwidth. For most gyrokinetic simulations of interest over 90% of the

runtime is spent in transferring data from the grid to the particles or vice-versa.

One cause of memory bandwidth constraint is inter-processor communication. On any given processor there is grid data and particle data. There are two ways these data sets intercommunicate on the CM-II. The first is a scatter with add which accumulates the particle charges onto the grid. To reduce the number of data packets being sent through the hypercube network, the router automatically sums charge contributions with the same target grid point as the data traverses each processor in the CM. The second way in which the particle and grid data communicate with each other is through a backtrace gather operation which updates the particle position. As the particle traverses across the computer in search of the grid cell to update its position, it leaves a message in each intermediate processor which will speed up the particle's ability to backtrace its path. Once the particle finds the processor with the target grid data it brings this piece of data back to its processor of origin by backtracking through the messages left in intermediate processors.

As the particle position values drift away from the grid cells where they were initialized, the inter-processor communication route for each particle increases. One method of avoiding this problem is to keep the particles semi-sorted shortening the communication paths. Each particle traveling the shorter route to obtain or deposit its target data provides a 40% overall speed-up in runtime. Also, since the particle routing uses memory, shorter paths allow us to dedicate more memory to particle information rather than router tracing. In fact, one is able to use twice as many particles when the semi-sort is employed.

When implementing any algorithm on a distributed memory MIMD machine like the Intel iPSC/860 the first and foremost question is the distribution of tasks among the various processors. Memory constraints require that each processor have only a small subset of the total data, so the most natural breakdown is to distribute the data across the processors. The precise breakdown must be one that balances the load evenly among the processors in the two main phases of the calculation, the particle-grid interaction and the FFT for the grid solve.

The nature of the particle-grid interaction is that several grid points near a particle must be referenced/modified. All of the particles near a processor boundary will need to access grid data from the neighboring processor, so it makes sense to transfer all of the grid data that will be needed just once and store redundant copies of strips around the interface. This saves both because any grid data sent will be sent exactly once, and because we may send the data in a large block and the communication hardware is more efficient for large blocks than small.

When doing any of the one dimensional FFT's that make up a three dimensional FFT it is convenient to have the whole line of the matrix in memory on a single processor, but of course this cannot be maintained for the whole FFT. If there are N_p grid points in each direction and N_p processors we may break up the computational box into N_p slabs of thickness N_p/N_p with two dimensions stored contiguously on each processor and the third broken across processors. There will be $O(N_p \log_2 N_p)^2$ floating point operations in the FFT and $O(N_p)^2$ floating point communications, so we expect a small communication to computation ratio.

With the data broken down in this way each time step may proceed in much the same way as on a serial machine with the exception of a more complex FFT and some bookkeeping to handle the redundant grid data near the processor boundaries. At the end of each time step when the particles have moved it is a simple operation to send particles that have crossed a line that indicates the edge of a processor's domain to its new processor. The whole process may then be repeated to integrate the system forward in time.

The iPSC/860 and CM-II are shown to each perform at about 5μ sec per particle per time step with the main difference being the CM-II's ability to perform an in-core simulation of 16 million particles on a 512 by 512 by 32 grid. The iPSC/860 could only run about a quarter this size. We plan to report the code's performance on the CM200 and the Intel Delta.

Derivation of Implicit Difference Schemes

by the Method of Differential Approximation

E. J. Caramana, Group X-1, LANL, Los Alamos, NM 87545

In this paper we use the continuum differential approximation of a time discrete difference scheme to show how one can derive simple operators that approximate fully implicit time differencing of a system of partial differential equations. This type of approximation has been considered before, and has been called "semi-implicit" [1]. It generally involves adding implicitly and subtracting explicitly some simple operator whose matrix form is easy to invert. This modifies the truncation error of the underlying difference scheme and, as far as time error is concerned, results in unconditional linear stability just as for a fully implicit scheme, but for much less work. The question of accurate reproduction of the original continuum system of equations must of course still be considered, however, numerical stability is assured. We show how the method of differential approximation [2,3,4] can be used to derive semi-implicit operators for any given system of equations. In this way we show explicitly how such operators arise from the truncation error of a "brute force" fully implicit differencing in time. This sheds light on the fundamental meaning of implicit differencing.

In the method of differential approximation one analyzes a finite difference scheme by converting it to a continuum set of equations by a simple Taylor series expansion keeping an "appropriate" number of truncation error terms. This, the differential approximation, represents the finite difference scheme and shows how it corresponds to the original system to be solved. It is a continuous system of equations and thus allows a direct comparison of the finite difference scheme to the original "primitive" system of equations. The method of differential approximation can be used to study the effects of nonconstant and nonlinear coefficients [2]. In the case of constant coefficients a direct correspondence with the usual Fourier series analysis of difference schemes can be established [3]. Often, instability of an unstable difference scheme will show up as ill-posed diffusion in the lowest order terms of the differential approximation. Some algebraic manipulation may be required to see this [2,4].

The answer to the question of how high in degree truncation error terms retained in the differential approximation to a finite difference scheme should be is easily ascertained by the following consideration. For a differential equation to faithfully represent the continuum limit of a discrete difference equation it must require the same initial and boundary data as the difference equation, that is, it must be of the same degree in all of its derivatives. If the difference equation, and thus its differential approximation, is of higher degree than the primitive differential equation it is to represent (usually this is done to obtain a higher order of approximation) then extra numerical boundary or initial conditions are needed. Such schemes will contain spurious computational solutions that have nothing to do with the primitive differential equation. These solutions may influence the stability and accuracy of the difference scheme and may explain why formally higher order difference schemes in some instances give poor results. Since the differential approximation to the difference scheme, as defined above, contains the additional computational solutions, it gives a way of analyzing their properties. Indeed, in this case the differential approximation will contain the primitive differential equation as a factor.

Consider the scalar, one-dimensional, second degree wave equation written as the coupled first order system

$$\frac{\partial u}{\partial t} = c \frac{\partial v}{\partial x}, \quad (1a)$$

$$\frac{\partial v}{\partial t} = c \frac{\partial u}{\partial x}, \quad (1b)$$

where c is the wave speed. A "brute force" fully implicit first order accurate time differ-

encing of Eqs.(1) can be written as

$$\left(\frac{\partial u}{\partial t} \right)^{n+1/2} = c \left(\frac{\partial v}{\partial x} \right)^{n+1}, \quad (2a)$$

$$\left(\frac{\partial v}{\partial t} \right)^{n+1/2} = c \left(\frac{\partial u}{\partial x} \right)^{n+1}, \quad (2b)$$

where the superscripts have the same meaning as before and the second order accurate, centered spatial differencing has once again been suppressed. Since we require that the degree of the difference scheme and its differential approximation agree, only terms of the form $\partial^2/\partial t \partial x$ may be added to Eqs.(2). Taylor expanding the rhs of Eqs.(2) about the $n + 1/2$ time level gives as its differential approximation

$$\frac{\partial u}{\partial t} - \frac{c \Delta t}{2} \frac{\partial^2 v}{\partial t \partial x} = c \frac{\partial v}{\partial x}, \quad (3a)$$

$$\frac{\partial v}{\partial t} - \frac{c \Delta t}{2} \frac{\partial^2 u}{\partial t \partial x} = c \frac{\partial u}{\partial x}. \quad (3b)$$

Next we rewrite Eqs.(3) in an equivalent but decoupled form. To this end we time differentiate Eq.(3b) and using Eq.(3a) obtain,

$$\left(1 - \frac{c^2 \Delta t^2}{4} \frac{\partial^2}{\partial x^2} \right) \frac{\partial^2 v}{\partial t^2} - c^2 \Delta t \frac{\partial^2}{\partial x^2} \left(\frac{\partial v}{\partial t} \right) = c^2 \frac{\partial^2 v}{\partial x^2}. \quad (4)$$

But this equation can then be rewritten as the coupled system

$$\frac{\partial u}{\partial t} = c \frac{\partial v}{\partial x}, \quad (5a)$$

$$\left(1 - \frac{c^2 \Delta t^2}{4} \frac{\partial^2}{\partial x^2} \right) \frac{\partial v}{\partial t} = c \frac{\partial u}{\partial x} + c^2 \Delta t \frac{\partial^2 v}{\partial x^2}. \quad (5b)$$

Equations (5) are exactly equivalent to Eqs.(3), that is, they represent an approximation to an implicit differencing scheme in the lowest order terms which comprise the differential approximation. Notice that if we had used the primitive system, Eqs.(1), to directly transform the mixed derivatives in Eqs.(3) we would only have obtained the lowest order diffusion terms and would have missed the essential term $\propto \Delta t^2$ in Eq.(5b).

We now wish to dwell on some properties of Eqs.(5). First, note that an explicit time differencing of Eqs.(1) [$n + 1 \rightarrow n$ in Eqs.(2)], known to be numerically unstable, also would lead to Eqs.(5) except that the sign of the dissipative term on the rhs of Eq.(5b) would be negative, indicating instability as ill-posed diffusion. However, if one drops the dissipative term in Eq.(5b) then Eqs.(5) are still unconditionally stable and represent a second order accurate approximation to Eqs.(1), reminiscent of a time centered differencing. To see that Eqs.(5) lead to an unconditionally stable scheme consider them differenced by the time staggered leapfrog scheme with the dissipative term dropped. This gives

$$u^{n+1/2} - u^{n-1/2} = c \Delta t \left(\frac{\partial v}{\partial x} \right)^n, \quad (6a)$$

$$v^{n+1} - v^n = c\Delta t \left(\frac{\partial u}{\partial x} \right)^{n+1/2} + c^2 \frac{\Delta t^2}{4} \left(\frac{\partial^2 u}{\partial x^2} \right)^{n+1} - c^2 \frac{\Delta t^2}{4} \left(\frac{\partial^2 v}{\partial x^2} \right)^n \quad (6b)$$

The last two terms on the rhs of Eq.(6b) are the usual "semi-implicit" terms—added implicitly and subtracted explicitly. We see that they naturally arise as a piece of the implicit differenced form, Eqs.(2), given directly by Eq.(6), in fact, as the only essential part. Leapfrog time differencing on Eqs.(1) yields $c^2 k^2 \Delta t^2 \leq 4$ as a stability criterion. Equations (6) modify this constraint to be

$$\frac{k^2 c^2 \Delta t^2}{1 + \frac{k^2 c^2 \Delta t^2}{4}} \leq 4 \quad (7)$$

which is readily seen to always be satisfied independently of Δt . This brings out the important point of implicit difference schemes; namely, they stabilize mainly by means of phase error, diffusion is subsidiary. That is, an implicit difference scheme comprises all time eigenvalues of a system of equations to be inside a radius of $1/\Delta t$ in the complex $1/t$ plane. This always results in a "stabilizing denominator", or in physics terms, a k dependent mass. The point is that the semi-implicit approximation does the same thing, only Eqs.(6), being more diagonal, are easier to invert than the fully implicit scheme given by Eqs.(2). The diffusion term in Eq.(6b) may be kept to damp unresolved spatial scales or to compensate the ill-posed diffusion that naturally arises from nondiffusive schemes applied to equations with nonconstant coefficients; if so, it should be treated implicitly to avoid a stability restriction due to time differencing of the differential approximation itself. However, a controlled amount of dissipation can always be added in any case. The differential approximation is seen to be "born implicit". The scheme used to difference it must only be conditionally stable in time with respect to the primitive system of equations for the resulting numerical scheme to be unconditionally, linearly stable. What we have shown is that since implicit time differencing compresses eigenvalues anyway, one can just as well do this from the outset by modifying the primitive system of equations appropriately before any discretization is applied (contrast Eqs.(5) and Eqs.(1)).

The vector analog of the coupled scalar wave equations, Eqs.(1), can be written as

$$\frac{\partial \vec{u}}{\partial t} = -c \nabla \times \vec{v} \quad (8a)$$

$$\frac{\partial \vec{v}}{\partial t} = c \nabla \times \vec{u} \quad (8b)$$

Following the same steps as before we find as the analog of Eqs.(5) for the differential approximation to a fully implicit time differencing of Eqs.(8) the result

$$\frac{\partial \vec{u}}{\partial t} = -c \nabla \times \vec{v} \quad (9a)$$

$$\left(1 + \frac{c^2 \Delta t^2}{4} \nabla \times \nabla \times \right) \frac{\partial \vec{v}}{\partial t} + c^2 \Delta t \nabla \times \nabla \times \vec{u} = c \nabla \times \vec{u} \quad (9b)$$

Again dropping the dissipative term linear in Δt and also considering a solenoidal \vec{v} field, we see that the term to be added implicitly and subtracted explicitly to the rhs of Eq.(8b), given any conditionally stable differencing approximation to Eqs.(8), is simply $c^2 \Delta t \nabla \times \vec{u}$.

The resulting scheme will be unconditionally stable and is completely analogous to that given in Eqs.(6). The order of the scheme used to difference the differential approximation to a primitive system should not be higher than that of the differential approximation itself, else that higher order scheme should be used to construct the differential approximation.

Finally, consider the MHD equations for the magnetic field \vec{B} and the velocity field \vec{v} written in a simplified form with the pressure neglected and all diagonal terms dropped. This system becomes

$$\frac{\partial \vec{B}}{\partial t} = \nabla \times (\vec{v} \times \vec{B}) \quad (10a)$$

$$\frac{\partial \vec{v}}{\partial t} = -\vec{B} \times (\nabla \times \vec{B}) \quad (10b)$$

We implicitly difference this nonlinear system by placing only \vec{v} and $\nabla \times \vec{B}$ on the rhs of Eqs.(10a) and (10b), respectively, at the $n+1$ time level. Then expanding about time $n+1/2$ time level yields as the differential approximation

$$\frac{\partial \vec{B}}{\partial t} - \frac{\Delta t}{2} \nabla \times \left(\frac{\partial \vec{v}}{\partial t} \times \vec{B} \right) = \nabla \times (\vec{v} \times \vec{B}) \quad (11a)$$

$$\frac{\partial \vec{v}}{\partial t} + \frac{\Delta t}{2} \vec{B} \times \left(\nabla \times \frac{\partial \vec{B}}{\partial t} \right) = -\vec{B} \times (\nabla \times \vec{B}) \quad (11b)$$

Using Eq.(11a) in (11b) and dropping terms linear in Δt yields

$$\frac{\partial \vec{B}}{\partial t} = \nabla \times (\vec{v} \times \vec{B}) \quad (12a)$$

$$\frac{\partial \vec{v}}{\partial t} + \frac{\Delta t^2}{4} \vec{B} \times \nabla \times \nabla \times \left(\frac{\partial \vec{v}}{\partial t} \times \vec{B} \right) = -\vec{B} \times (\nabla \times \vec{B}) \quad (12b)$$

This gives $(-\Delta t/4) \vec{B} \times \nabla \times \nabla \times (\vec{v} \times \vec{B})$ as the term to be added implicitly (only respect to \vec{v} ; \vec{B} is taken as known at the n time level) and subtracted explicitly from the rhs of Eq.(12b). This is the semi-implicit MHD term originally derived by Harned.

In summary, it has been shown how the method of differential approximation can be used to construct simplified implicit schemes that are unconditionally linearly stable so that accuracy is the only remaining consideration. This has been illustrated by a series of worked examples. The number of terms to be kept in the differential approximation to any difference scheme has been given a unique definition. This is that the differential approximation require no more initial and boundary data for its solution than does the difference scheme. By use of the method of differential approximation the precise relation of semi-implicit time differencing as an approximation to implicit differencing is transparent. In addition, how implicit time differencing results in stability is also clarified

REFERENCES

1. D. D. Schnack, D. C. Barnes, Z. Mikic, D. S. Harned, and E. J. Caranahan, *J. Comput. Phys.* **70**, 330 (1987).
2. C. W. Hirt, *J. Comput. Phys.* **2**, 339 (1968).
3. R. F. Warming and B. J. Hyett, *J. Comput. Phys.* **14**, 159 (1974).
4. Yu. I. Shokin, *The Method of Differential Approximation* (Springer-Verlag, New York, 1983).

14ème Conférence Internationale sur les Simulations Numériques des Plasmas
à Annopolis (Maryland) du 3 au 6 Septembre 1991

PLASMA EXPANSION INTO VACUUM

S. BOUQUET
*Commissariat à l'Energie Atomique CEV-M BP7
77181 Courcy France*

G. MANFREDI
M.R. FEIX
*CNRS-PMMS 3A, Av. de la Recherche Scientifique
45071 Orléans Cedex France*

In this paper, we study the 1-dimensional expansion of a two components plasma (ions + electrons) in plane geometry. The plasma is initially at rest with various temperatures T_i and T_e . For large value of the ratio kT_i/kT_e (U : coulomb energy of the plasma) and since the electron mass is much smaller than the ion mass, the electron gas expands while the ions are still at rest. On the other hand, from analytical considerations, it can be shown that the asymptotic evolution (time $t \rightarrow +\infty$) corresponds to a locally neutral plasma expanding as a free particle. As a consequence, a transient solution exists in which the self-coherent electrostatic field brings together the two species.

The study of the entire evolution is performed numerically by solving the Vlasov-Poisson system in two different frames. The first one is the regular Eulerian space (x, t) and the second one is a rescaled space (\hat{x}, \hat{t}) . In this new space, \hat{x} and \hat{t} are respectively given by

$$\hat{x} = x/(1 + \Omega t), \quad \hat{t} = t/(1 + \Omega t),$$

where Ω is a free positive parameter. With such a rescaling, no friction appears and, the phase space element volume and the distribution function are preserved. Moreover, this rescaling absorbs the final expansion of the plasma and as a consequence, we do not have to rebuild the phase space grid to take into account the motion of the plasma.

Finally, the time \hat{t} is renormalized, i.e., \hat{t} moves in the range 0 to $1/\Omega$ for t going from 0 to $+\infty$. The price to pay is that for reaching the limit time $1/\Omega$, we have to take smaller and smaller time steps. In the rescaled space, the asymptotic state is consequently obtained in a finite time and it corresponds to a neutral gas frozen by the renormalization at time $\hat{t} = 1/\Omega$. In the regular (x, t) frame, we have a free expansion of a neutral gas with temperature going to zero.

As an extension, some considerations are given in the case of a spherical geometry.

// S. BOUQUET, L. CAIRO and M.R. FEIX J. Plasma Physics (1985), 34, 127.

Thursday Afternoon

September 5, 1991

Panel Discussion
2:00pm - 3:30pm

Regatta Ballrooms A & B

The Transition to UNICOS and Distributed Computing: What's in it for me?

Panel Members:

Arthur Bland, Martin Marietta
Bob Ferraro, JPL
Dave Forslund, LANL
Richard Zwakenberg, NERSC
Barry Howard, NERSC, Moderator

Theme

Several national supercomputer centers have either already made or are in the process of making the transition from CTSS to UNICOS. This transition is making a significant impact on the modus operandi of both the users and the providers of computing services. Representatives of these two groups are serving on the panel and they will lead a discussion centered around

- Tips on how to make the transition to UNICOS easier.
- The benefits of moving to a more distributed computing environment, which includes UNICOS supercomputers, and how to take advantage of these benefits.

Thursday Afternoon
September 5, 1991

Poster Session (PT)
3:30pm - 5:30pm

Regatta Ballroom C

VLASOV-MAXWELL ALGORITHM FOR ELECTROMAGNETIC PLASMA SIMULATION ON DISTRIBUTED ARCHITECTURES

Philip M. Campbell Scott A. von Laven
Cray Research, Incorporated Mission Research Corporation
6565 America's Parkway, Suite 830 1720 Randolph Road, SE
Albuquerque, NM 87110 Albuquerque, NM 87106

Extended Abstract for the
14th International Conference on the Numerical
Simulation of Plasmas
03-3 September 1991

ABSTRACT

A Vlasov-Maxwell (V-M) algorithm for electromagnetic plasma simulation will be described and its features compared with the more familiar particle-in-cell (PIC) algorithm. Issues that arise in implementing each of these algorithms on distributed memory multiprocessors will be discussed. Computational results will be presented for the Connection Machine (CM-2), including computational efficiency and scalability. We expect that the greater computational efficiency of V-M on massively parallel machines will alter the perception from serial computing that V-M is only appropriate for a restricted class of problems.

1 BACKGROUND

The most popular methods for electromagnetic plasma simulation are based on the particle-in-cell (PIC) algorithm. The PIC algorithm solves Maxwell's equations on a spatial grid with the necessary charge and current densities obtained by following the motion of individual charged particles. Currents are found by interpolating particle quantities to the grid (scatter phase), and field values at individual particle positions are found from a reverse interpolation (gather phase). The PIC algorithm consists of two distinct data structures: one for the particles and one for the fields. This dual data structure presents a special challenge to parallelization on distributed architectures.

On the hypercubes (course-grained MIMD machines) the PIC data structure requires dynamic load balance which can be expensive in terms of communications overhead (Refs. 1, 2, 3). On the Connection Machine (and other fine-grained SIMD machines), nonuniform particle distributions involve even greater communications cost in the gather and scatter phases (Refs. 4, 5). Even on vector machines such as the Crays, the PIC algorithm requires special attention in the gather/scatter routines to achieve any degree of vectorization (Refs. 6, 7).

Our experience in mapping the dual data structures of PIC onto the hypercubes and Connection Machine confirms that PIC is not an ideal algorithm for distributed memory architectures. This leads us to consider an alternate approach to plasma simulation, the Vlasov-Maxwell (V-M) algorithm (Refs. 8, 9). The V-M algorithm involves finite differencing of the particle distribution function in phase space. Whereas PIC is Lagrangian in its treatment of the particle distribution, V-M is Eulerian and thereby avoids the dual data structure.

For many problems on serial machines, PIC is computationally more efficient (Ref. 10). The V-M approach is normally used in situations where the plasma phase-space distribution is particularly simple. There are some types of problems where PIC fails or is computationally impractical. In these cases V-M is the preferred algorithm.

Because of the differing communications requirements on massively parallel machines, the relative advantages of PIC and V-M must be reconsidered. The complexity and overhead associated with load balancing on hypercubes is eliminated with V-M. The communications requirements for V-M on the CM-2 can be handled by the NEWS network, thereby avoiding the greater cost of global communication. On the other hand, V-M involves more memory and more computation than PIC. There is also an additional stability condition to which PIC is not subject. In this paper, test calculations quantifying these trade-offs will be presented.

2. THE VLASOV-MAXWELL FORMULATION

In the Vlasov approach (Refs. 8, 9) the particle distribution in a plasma (e.g., a particle beam) is described by its density in phase space $f(\vec{x}, \vec{v}, t)$. The distribution function is a continuum representation of the particle distribution where

$$f(\vec{x}, \vec{v}, t) d^3x d^3v = \text{number of particles in } d^3x \text{ at } \vec{x} \\ \text{with velocities in } d^3v \text{ at } \vec{v}. \quad (1)$$

The distribution function obeys the collisionless Vlasov equation

$$\frac{\partial f}{\partial t} + \sum v_i \frac{\partial f}{\partial x_i} + \sum a_i \frac{\partial f}{\partial v_i} = 0 \quad (2)$$

where the acceleration of a particle in an electromagnetic field is

$$a_i = \frac{e}{m} \left[E_i + \frac{1}{c} (\vec{v} \times \vec{B})_i \right] \quad (3)$$

The electric and magnetic fields are solved by Maxwell's equations as in conventional PIC.

The charge and current densities are obtained by

$$\rho = e \int f d^3v \quad (4)$$

One technique for solving the Vlasov equation is by finite differences. Phase space is discretized by a grid. Thus, for two-dimensions: $x(i) = i\Delta x$, $y(j) = j\Delta y$, $v_x(k) = k\Delta v_x$, and $v_y(l) = l\Delta v_y$. The spatial grid is the same as used in solving Maxwell's equations, so the sums over velocity space in the discrete version of Equation 4 give currents correctly indexed to the spatial grid for the field solution.

3 RESULTS OF MAPPING THE VLASOV ALGORITHM ONTO THE CONNECTION MACHINE

A 2D plasma simulation code using the V-M algorithm has been implemented on the CM-2. A test problem consisting of a 32×64 spatial grid with a 16×32 velocity grid maps onto a quadrant of the CM-2 with 32 velocity components stored locally in virtual processors and the rest of the grid (16,384 cells) distributed over the 16K-node machine.

Since the spatial gridding of the particle distribution conforms to the gridding for the fields, gather/scatter operations require only gridwise (NEWS) communication. This type of communication, VP-to-VP or along hypercube wires, is much faster on the CM-2 than general communication using the Router. A two-stream instability test problem executed on one quadrant of the CM-2 at over 200 Mflops. By comparison the corresponding PIC simulation achieved only 10 Mflops.

Applications other than the two-stream instability will also be discussed. These include relativistic high-power microwave sources and intense electron diodes. The diode problem involves multiple plasma components, all of which can be treated within the V-M framework.

4. REFERENCES

1. Liewer, P., and Decyk, V., "A Universal Concurrent Algorithm for Plasma Particle-In-Cell Simulation Codes," Fourth Conference on Hypercube, Concurrent Computers and Applications, Monterey, CA, March 1989.
2. Walker, D. W., "The Parallel Implementation of a Large-Scale Particle-In-Cell Simulation Code," submitted to *Concurrency: Practice and Experience*, (1990).
3. Campbell, P. M., Carmona, E. A., and Walker, D. W., "Hierarchical Domain Decomposition with Unitary Load Balancing for Electromagnetic Particle-In-Cell Codes," Fifth Distributed Memory Computing Conference, Charleston, SC, April 1990.
4. Campbell, P. M., Campbell, M. M., and von Laven, S. A., "Performance of 2D and 3D Electromagnetic Particle-In-Cell Codes on Parallel Architectures," CIBP-1-MRC-062/MRC/ABQ-R-1241, September 1990.
5. Walker, D. W., to be provided.
6. Nishiguchi, A., Orii, S., and Yabe, T., "Vector Calculation of Particle Code," *Journal of Computational Physics* 61, (1985).
7. Horowitz, E. J., "Vectorizing the Interpolation Routines of Particle-in-Cell Codes," *Journal of Computational Physics* 68, (1987).
8. Byers, J. A., and Killeen, J., "Finite Difference Methods for Collisionless Plasma Simulation," *Methods of Computational Physics*, eds. Fernbach and Rotenberg, (Academic Press, New York, NY), Vol. 9, p. 259, (1970).
9. Armstrong, T. P., Harding, R. C., Knorr, G., and Montgomery, D., "Solution of Vlasov's Equation by Transform Methods," *Methods of Computational Physics*, eds. Fernbach and Rotenberg, (Academic Press, New York, NY), Vol. 9, p. 29 (1970).
10. Denavit, J., and Krueer, W. L., "Comparison of Numerical Solutions of the Vlasov Equation with Particle Simulations of Collisionless Plasmas," *Phys. Fluids* 14, 1782 (1971).

A FULLY PARALLEL 2D VLASOV CODE Comparison Between Connexion Machine and Cray H versions.

E. FIJALKOW*, E. JAMIN*, M.R. FEIX*
F. BERTRAND**, A. GHIZZO**, M. SHOUCRI**

* PMMS/CNRS - 3A av de la Recherche Scientifique - 45071 Orléans Cédex 2 - France
** Labo de Phys Théor. - BP 239 - Univ. Nancy I - 54506 Vandœuvre-Les-Nancy - France
*** IRS-Energie, IR Hydro-Québec - 1800 Montée Ste-Julie - Varinnes, Québec - Canada J0L2P0

We present here a method to solve the 2D Vlasov equation in phase space (2 space dimensions and 2 velocity dimensions). After that, we shall compare the connexion machine version, written in CM Fortran (subset of Fortran 90) and the Cray II version, written in CFT 77, an extension of Fortran 77. We shall compare realizations on a CM2, and Cray II and execution times on these machines

1 - THE VLASOV CODE IN 2D

The general Vlasov equation in 2D can be written in Cartesian coordinates :

$$\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} + v_y \frac{\partial f}{\partial y} + (E_x - B_y v_y) \frac{\partial f}{\partial v_x} + (E_y + B_x v_x) \frac{\partial f}{\partial v_y} = 0 \quad (1)$$

That equation is subjected to initial conditions :

$$f(x, y, v_x, v_y, t=0) = g(x, y, v_x, v_y) \quad (2)$$

and boundary conditions :

always zero for $v > v_{max}$ in x and/or y
periodicity or free boundaries in x and y .

The terms in brackets are the x and y components of the Lorentz force. E and B are given in the general case by Maxwell equations. In the case of the actual program, B is taken as a constant, so that these equations are reduced to the Poisson equation: (3) plus a constant in time velocity dependant term :

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) E(x, y, t) = \rho(x, y, t) / \epsilon_0 \quad (3)$$

Where the density ρ is given by :

$$\rho(x, y, t) = \iint \dots \int f(x, y, v_x, v_y, t) dv_x dv_y - N_0 \quad (4)$$

and N_0 is the ions background density.

On way to solve numerically the system, made by equations (1) and (4), we look for a centered scheme. This is done by the fractional steps method. Eq (1) breaks up to four independent steps.

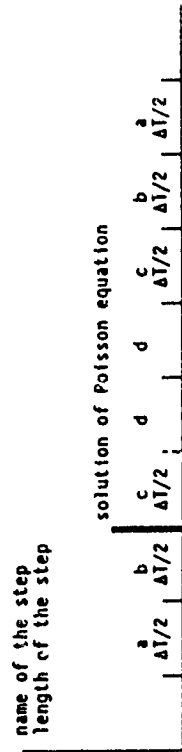
$$\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} = 0 \quad (5a)$$

$$\frac{\partial f}{\partial t} + v_y \frac{\partial f}{\partial y} = 0 \quad (5b)$$

$$\frac{\partial f}{\partial t} + (E_x - B_y v_y) \frac{\partial f}{\partial v_x} = 0 \quad (5c)$$

$$\frac{\partial f}{\partial t} + (E_y + B_x v_x) \frac{\partial f}{\partial v_y} = 0 \quad (5d)$$

The numerical scheme takes the form given by fig 1.



2 - NUMERICAL SOLUTION PRINCIPLE

From the Vlasov equation (1) we see that the total mass of phase space fluid in a given cell is invariant in time :

$$M = \int \int f(x, y, v_x, v_y, t) dx dy dv_x dv_y \quad (6)$$

We are looking for the numerical solution of eq (1) on a regular grid $\Delta x, \Delta y, \Delta v_x, \Delta v_y$, each cell centered on the point (i_x, i_y, j_x, j_y) . The change of the fluid mass in a given cell comes only from the quantity of fluid crossing the boundary in a given step. Let us look for example at the solution of eq (5a), for the i_x, i_y, j_x, j_y cell at time step t_n :

$$\delta^- M_{i,j,j} = \int_{v_x - \Delta v_x}^{v_x + \Delta v_x} \int_{v_y - \Delta v_y}^{v_y + \Delta v_y} f_{i,j,j}^{n+1/2} dv_x dv_y - \int_{v_x - \Delta v_x}^{v_x + \Delta v_x} \int_{v_y - \Delta v_y}^{v_y + \Delta v_y} f_{i,j,j}^n dv_x dv_y \quad (7)$$

In way to solve this equation we approximate f linearly in the cell :

$f(x, y, v_x, v_y, t_n) = A_1 v_x^2 + A_2 v_y^2 + B_1 v_x + B_2 v_y + C$ the coefficient, A_1, A_2, B_1, B_2 and C being calculated from neighbour values of f .

If we write down the value of f at the grid points as $f(i_x, i_y, j_x, j_y)$, the value of the function in the cell (i_x, i_y, j_x, j_y) is given by :

$$f(x, y, v_x, v_y) = \frac{1}{2\Delta x} [f(i_x + 1, j_y, j_z, j_t) + f(i_x - 1, j_y, j_z, j_t)](x - x_i) +$$

$$+ \frac{1}{2\Delta y} [f(i_x, j_y + 1, j_z, j_t) + f(i_x, j_y - 1, j_z, j_t)](y - y_i)$$

$$+ \frac{1}{2\Delta v_x} [f(i_x, j_y, j_z + 1, j_t) + f(i_x, j_y, j_z - 1, j_t)](v_x - v_{ix})$$

$$+ \frac{1}{2\Delta v_y} [f(i_x, j_y, j_z, j_t + 1) + f(i_x, j_y, j_z, j_t - 1)](v_y - v_{iy})$$

$$+ f(i_x, j_y, j_z, j_t)$$

Introducing this last result into eq(7) we obtain the quantity of space fluid lost by the i_x cell to the $(i_x + 1)$ cell.
Total phase space fluid balance for the i_x cell take into account the gain of fluid mass from $(i_x - 1)$ cell and the fluid lost by the i_x cell.

$$\delta M_{i_x} = -\delta^- M_{i_x} - 1 + \delta^- M_{i_x}$$

δx , the one dimensional shift length is given from equation (5a) for half a time step as:

$$\delta x_j = v_{jx} \Delta T/2$$

To solve eq(5b) to (5d) the same process is used, the equivalent of eq(7) being calculated for shifts that are respectively :

$$\delta y = v_{jy} \Delta T/2, \quad \delta v_x = (E_x - B_y v_j / \Delta T/2)$$

$$\delta v_y = (E_y + B_x v_j) \Delta T$$

(in agreement with scheme of fig (1)).

3 - PRACTICAL IMPLEMENTATION ON THE CONNEXION MACHINE

The implementations use here the Fortran 90 of CMF. To be efficient, the arrays need to be temporary 4d arrays. This is done by the SPREAD functions. The version of the code presented here being periodic in x and y , the shifts in these directions are achieved by call to the CSHIFT function (for eq (5c) by a call to EOSHIFT).

$$FT = \text{CSHIFT}(F, 1, \text{SPREAD}(\text{SPREAD}(IXX, 2, \text{IREPY}), 1, NY + 1))$$

This is equivalent to the calculation of FT in the Cray version, but for all four dimensions.

FT = SPREAD (SPREAD (SHVX, 2, REPY), 1, NY + 1), 1, NX + 1)*
&(FT + SPREAD (SPREAD (SPREAD (AX, 2, IREPY), 1, NY + 1), 1, NX + 1))*
&(SHVFT (FT, 1, 1) - CSHIFT (FT, 1, -1)))

Executes the calculus of DM in the Cray version, again in 4d.

$$F = F - FT + \text{CSHIFT}(FT, 1, -1)$$

Final result.

4 - EXECUTION TIMES

Only preliminary results are at present available. More results shall be presented at the conference. For example, a $16 \cdot 16 \cdot 64 \cdot 64$ problem.

Computer	Number of processors	Precision	Time per time step	Time per cent on CM
CM 2	8K	32 bits	3,77 sec	98%
	16K	32 bits	2,15 sec	98%
	8K	64 bits	14,06 sec	100%
	16K	64 bits	7,54 sec	99%
CM 2a	4K	64 bits	2,5 sec	89%
Cray II	1	64 bits	3,65 sec	

5. COMPARISON WITH PREVIOUS WORK.

We have compared our results with these of M. Shourci (IEEE Trans on Plasma Science vol P5-7 n°2, p 69-72, 1979) in the case of the nonlinear evolution of a monochromatic wave. The initial condition is

$$g(x, y) = \frac{1}{2\pi} \exp(-(\nu_x^2 + \nu_y^2)/2)(1 + \alpha \cos k_x \cos k_y)$$

with $k_x = k_y = 3/\sqrt{2}$, $\alpha = .1$. The results are equivalent.

Actually we work on a new version of Poisson Solver in way to improve the program.

CONCLUSION

To conclude, we can say that the implementation of the code on the CM is very efficient, specially due to the Fortran 90 features. On the other hand, for stability purpose, we need 64 bits precision, so that we have to use a CM2a machine. The Cray II computer on the other hand, because of his large memory allow to run more realistic versions of the program.

To finish, it is clear that the Connexion Machine is a good test for algorithms to be used on futur massively parallel systems.

Two applications of the Connection Machine: The 3-d Ginzburg-Landau Equation and A 1-d Electrostatic Plasma Simulation

G. Furnish
Institute for Fusion Studies
The University of Texas at Austin

L. Gil
I N L N , U M R C N R S 129
Université de Nice Sophia Antipolis

M. J. LeBrun
Japan Atomic Energy Research Institute
Naka Fusion Research Establishment

The Connection Machine (CM) provides several thousand processors which execute simultaneously in synchronous fashion. This architecture implements a programming model known as single instruction, multiple data stream, and permits easy expression of problems exhibiting a massive degree of parallelism. With some practice this intellectual paradigm can be productively applied to a wide range of problems, including the two discussed herein. In addition to the tremendous ease of expression of algorithms, we have found that the unique abilities of the Connection Machine also allow the real time visualization of results during the program execution.

1 The 3-d Ginzburg-Landau Equation

The Ginzburg Landau equation for a complex field A can be expressed as:

$$\partial_t A = (\mu_r + i\mu_i)A + (\beta_r + i\beta_i)|A|^2 A + (c_r + i c_i)\nabla^2 A$$

This equation arises in the description of the slow space and time variation of a field of nonlinear oscillators, and appears in all systems which undergo Hopf bifurcation. This nonlinear partial differential equation was solved computationally by finite differencing in time and space.

A regular spatial grid was divided into two interpenetrating meshes each having half the points, along any direction on the grid the even indexed points belonged to one mesh, and the odd points to the other. The field was advanced in a time step by an explicit predictive advance on one mesh, an implicit predictive advance on the other, an explicit corrective advance on the second mesh, and finally an implicit corrective advance on the first. This scheme is numerically stable and second order accurate in both space and time.

We first implemented this algorithm in two spatial dimensions. We programmed the Connection Machine in C*, a variant of ANSI C with extensions for declaring, allocating, and using parallel data structures. Each grid point was assigned to one CM processor, with the processors grouped into two meshes as described above. The algorithm was implemented in an index free manner, since all terms in the equation except the laplacian relate the new value to the old value at the same grid point. In C*, a typical update of the field quantities is:

```
a = b * c;

which in standard C on a standard architecture would be:

for(i=0; i < N; i++)
  for(j=0; j < M; j++)
    a(i,j) = b(i,j) * c(i,j);
```

To compute the laplacian, it is necessary to have the field values at the neighboring grid sites. This sort of deterministic message passing along grid axes is very efficient on the Connection Machine, and is implemented in a variety of syntases ranging from parallel indexing to run time library functions. We used the `get_from_torus()` function which is the fastest of all known methods, and which directly implements the periodic boundary conditions we desired for the system.

Having constructed the two dimensional code we then proceeded to extend the code to three spatial dimensions. To do this in C* on the Connection Machine required only the modification of the declarations of the parallel field variables. Since all computations were performed using the elemental array processing notation of C*, it was not necessary to modify any of the source code implementing the solution algorithm. Rather, we merely had to change four lines of code which defined the geometric configuration of the spatial grid to indicate the presence of a third grid dimension.

In contrast, a normal serial language would have required the addition of an extra index to every array reference, and the conversion of every double index loop over grid quantities into a triple loop. For our code this would have resulted in several hundred modifications to the solution algorithm and a commensurate opportunity for the accidental introduction of coding errors. It is clear then, that for this problem the expressive power of C* on the Connection Machine freed us from many of the mundane concerns of software methodology and verification, and allowed us to concentrate instead on the physics of interest. Converting the code from a two dimensional nonlinear PDE solver to a three dimensional one, took less than an hour.

In addition to the excellent applicability of the Connection Machine to the core solution algorithm, we also found the machine to be extremely well suited to the unique concerns of scientific visualization. In particular, we needed to be able to visualize the evolution of a complex scalar field in three dimensions. The quantities we wished to be able to view were the real and imaginary parts of the field A , and A_1 , and the modulus $|A|$. In the two dimensional problem it is possible to represent these quantities directly by dumping to the screen a color bitmap of the field quantity of interest, but the visualization in three dimensions is less obvious.

We found, however, that it was very easy to program the Connection Machine to pick out slices of the three dimensional field (each slice is itself a two dimensional field representing the full system extent in x and y and corresponding to a distinct z), and lay them out next

to each other in a regular pattern. In our case we were able to pick out every fourth slice from a 64 x 64 x 64 simulation volume and lay these slices out on a 256 x 256 grid for dumping to the display. This was expressed in four lines of C code:

```
ix = pcoord(0) % 64;
iy = pcoord(1) % 64;
iz = (pcoord(2) / 64) * 10 + (pcoord(0) / 64) * 4;
f = {ix}[iy][iz]*0;
```

Here the variables ix, iy and iz are three dimensional parallel index variables of the same shape and size as the field quantities (64 x 64 x 64). ix is the field quantity to be visualized, and y is the 256 x 256 array for dumping to the display. Notice that the last line accomplishes the work of moving all the requested data elements from the field to the display array. No loops were required. The v field was then dumped to the display at each timestep, allowing a continuous monitoring of the field evolution. The accompanying video illustrates the phenomena observed.

2 One Dimensional Electrostatic Plasma Simulation

With the experience gained in the above endeavors, we then turned to the simulation of an electrostatic plasma in one dimension. This well known program [1, 2] provides a simple context for experimenting with the application of massively parallel techniques to particle in cell simulations.

In a particle in cell code the advancement of each particle is independent of the advancement of any other particle, and is therefore naturally a parallel process. The common serialization over particles is an artificial complexity introduced by the architecture of serial processing machines. This cycle of endlessly complicating programs by serializing operations which naturally ought to be done in parallel is broken by the Connection Machine. Specifically, using the array processing notation of the CM, it is possible to write an index free pushing algorithm which simultaneously advances all particle positions and velocities. However, a central goal of this research was to see if it would be possible to write a program which took good advantage of different machine architectures. Since parallel machines are relatively scarce, it seems desirable to have programs which operate correctly on serial architecture machines, but which can also operate in parallel on the CM or similar machines. It is naturally undesirable, however, to express every algorithm in two code sections, one for serial machines and one for parallel machines, as this again introduces artificial complexity into the code and begs for errors to be made in algorithm implementation, etc.

These competing concerns were ameliorated through the use of the tools of the IFS Numerical Tokamak [3]. Specifically, the use of the Fortran preprocessor MPPL made it possible to code the algorithm using macros which expand to the correct syntax on whichever architecture machine the code is compiled on. The pushing step is coded as:

```
start_loop(1,1,0)
( code to interpolate the field, loop y results in ext )
vx_o(1) = vx_o(1)*ext
vtx = (col4 + vx_o(1) * col5 + vx_o(1) - ext + vx_o(1))
vty = (col10 + vx_o(1) * col11 + vx_o(1) * col12 + vx_o(1))
vtz = (col16 + vx_o(1) * col17 + vx_o(1) * col18 + vx_o(1))
```

```
vx_o(1) = vtx + ext
vy_o(1) = vty
vz_o(1) = vtz
x_o(1) = x_o(1) + dt * vx_o(1)
apply_bc(o)
end_loop(1)
```

On a normal serial computer, this code expands as a regular do loop over particles, using the familiar indexing for particle and field quantities. On the CM, however, this expands as a set of index free parallel array processing directives (start_loop() and end_loop()) expand to nothing, indices on particle quantities vanish, etc.). The apply_bc() macro expands to a normal serial expression or to a parallel expression, again depending on the target environment. The use of such macros makes it possible to take advantage of unique machine capabilities while maintaining portability.

The parallel capabilities of the CM also enhanced the diagnostic capabilities of our code. In particular, on the CM we were able to implement a phase space diagnostic which "accumulates" particles onto a 256 x 256 x-v grid for display. The number of particles in each cell in phase space is then represented in color. In this way it is possible to have density information on phase space plots, and to see the continuous evolution of the phase space distribution. These capabilities are sorely lacking in the traditional black and white still frame images which are typically plotted only a few times during a simulation run. The accompanying video which shows the interaction of opposing electron beams, and also the interaction of a cold beam with a warm plasma demonstrates the utility of this arrangement.

References

- [1] C. K. Birdall and A. B. Langdon. *Plasma Physics via Computer Simulation*. McGraw Hill, New York. 1985.
- [2] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. Adam Hilger, Bristol and New York. 1988.
- [3] M. J. LeBrun, M. G. Gray, G. Furnish, T. Tajima, and W. D. Nystrom *The Numerical "Laboratory" for Plasma Simulation: A Modern Approach for the Construction and Maintenance of a Large Simulation Code* Proceedings of the 14th International Conference on the Numerical Simulation of Plasmas

NUMERICAL SOLUTION OF THE RELATIVISTIC VLASOV EQUATION ON THE NRL CONNECTION MACHINE*

Glenn Joyce, Jonathan Kröll, and Eric Esarey

Beam Physics Branch
Plasma Physics Division
Naval Research Laboratory
Washington, DC 20375

The use of the wakefields induced in relatively high density plasmas ($n \sim 10^{14} - 10^{16}$) by high current relativistic electron beams (the plasma wakefield accelerator) and short pulse lasers¹⁻³ (the laser wakefield accelerator) has recently stimulated interest in the numerical simulation of relativistic beam-plasma interactions. Most of the numerical work in this area has been done by using particle simulation techniques⁴, but another approach is the direct numerical solution of the Vlasov equation⁵.

There were a number of attempts at the solution of the Vlasov equation in the 1960's and early 1970's^{6,7} but the limitations of computers in both memory and speed, the difficulties inherent in the use of expansion techniques, and the problem of representing phase space in more than one dimension led most researchers to abandon Vlasov equation solutions in favor of particle simulation. With the advent of massively parallel computers, and the large amount of memory available, it seems appropriate to revisit the Vlasov equation and determine whether or not it can be useful in the understanding of the physics of beam-plasma interactions in multiple dimensions. An attractive feature of the Vlasov simulation is the absence of "particle noise" which tends to obscure some of the features of the physics of these interactions.

We have begun our studies with solutions in one dimension⁸ (z, p_z) and are now in the process of extending the solutions to higher dimension. We will discuss the solutions in two spatial dimensions (r, z) and one phase space dimension (p_z) as a prelude to a full two dimensional solution of the Vlasov equation. (For laser-plasma interactions, some terms with θ dependence must be included, but for simplicity, we will omit these terms from our discussion.)

The Vlasov equation is

$$\frac{\partial f}{\partial t} + \frac{p_z}{m\gamma} \cdot \frac{\partial f}{\partial z} + e \left(E + \frac{p_z v}{m\gamma c} \right) \cdot \frac{\partial f}{\partial p} = 0, \quad (1)$$

where

f is the plasma electron distribution function,

p is the particle momentum,

and

$$\gamma = \sqrt{1 + (p_z^2 + p_\theta^2 + p_z^2)/m^2 c^2}.$$

We assume that the ions form a uniform background, and that the relativistic electron beam is either an external charge and current distribution, or that it can be represented as a second Vlasov species. We can define a reduced distribution function as

$$F(r, z, p_z) = \int dp_\theta dp_\theta f(r, z, p_r, p_\theta, p_z). \quad (2)$$

The reduced distribution function has some of the properties of a fluid representation of the plasma and some of the properties of a distribution function.

It is convenient for problems of this sort to change variables from the usual laboratory variables (r, z, t) to a "speed of light window" defined by the variables ($r, z, ct - z, \tau - t$). In these variables, the equation for the reduced Vlasov equation can be written in the form

$$\frac{\partial F}{\partial \tau} + c \frac{\partial}{\partial z} \left[\left(1 - \frac{p_z}{m\gamma c} \right) F \right] + \frac{\partial}{\partial p_z} \left(\frac{p_z}{m\gamma} F \right) + \frac{\partial}{\partial p_z} \left[\left(E_z + \frac{p_z p_\theta - p_\theta p_z}{\gamma m c} \right) F \right] = 0. \quad (3)$$

In order to obtain the radial components of momentum, we use a cold fluid approximation for v_r and the conservation of canonical angular momentum to recover p_θ . The cold fluid equation is obtained via further integrals of the reduced distribution function.

To complete the equations, we use Maxwell's equations which we write in the Lorentz gauge as

$$\nabla_1^2 \phi - \frac{1}{c} \frac{\partial}{\partial t} \left(\frac{1}{c} \frac{\partial \phi}{\partial t} + 2 \frac{\partial \mathbf{A}}{\partial t} \right) \phi = -4\pi \rho, \quad (4)$$

$$\nabla_1^2 \mathbf{A} - \frac{1}{c} \frac{\partial}{\partial t} \left(\frac{1}{c} \frac{\partial \mathbf{A}}{\partial t} + 2 \frac{\partial \phi}{\partial t} \right) \mathbf{A} = -\frac{4\pi}{c} \mathbf{j}, \quad (5)$$

$$\nabla_1 \cdot \mathbf{A} - \frac{1}{c} \frac{\partial \phi}{\partial t} - \frac{1}{c} \frac{\partial \mathbf{A}}{\partial t} = 0. \quad (6)$$

For efficient use of the Connection Machine, we use explicit, finite difference solutions of the above equations. Note that, in order to run efficiently on the Connection Machine, the layout of the coordinates may be as important as the algorithms which are chosen. In addition to keeping data movements as localized as possible, it is important not to have large numbers of processors idle during the calculations. We are still in the process of investigating various methods of representing these equations some of which are two dimensional and some of which are three dimensional.

In the (r, ζ, τ) coordinate system, all information flows to larger ζ . We can use this property to guide us in the differencing of the equations. The Maxwell equations we solve by standard techniques and will not discuss further. The Vlasov equation, is split into the equations

$$\frac{\partial F}{\partial \tau} + c \frac{\partial}{\partial \zeta} \left[(1 - p_z / \gamma mc) F \right] = 0, \quad (7)$$

$$\frac{\partial F}{\partial \tau} + \frac{\partial}{\partial r} \left(\frac{p_r}{\gamma} F \right) = 0, \quad (8)$$

$$\frac{\partial F}{\partial \tau} + \frac{\partial}{\partial p_z} \left[\left(p_z + \frac{p_r^2}{\gamma mc} - p_\theta^2 \right) F \right] = 0, \quad (9)$$

which are solved by interleaving the solutions of the individual equations into an equivalent solution of Eq. (3). Eq. (7) is solved by 1st order upwind differencing. Eq. (8) requires special attention and is differenced along with the cold fluid equation and Eqs (4) - (6) to recover an undamped oscillator solution in the linear limit. Finally, Eq (9) is differenced using an FCT algorithm to insure a positive solution⁹. The interleaving is selected to produce a first order solution to the reduced Vlasov equation.

References

- 1) T. Tajima and J. M. Dawson, Phys. Rev. Lett. 43, 267 (1979).
- 2) P. Chen, J. M. Dawson, R. V. Huff and T. Katsouleas, Phys. Rev. Lett. 54, 693 (1985)
- 3) P. Sprangle, E. Esarey, A. Ting and G. Joyce, Appl. Phys. Lett. 53, 2146 (1988)
- 4) R. Keinigs, M. E. Jones, and J. J. Su, IEEE Trans. Plas. Sci. PS-15, 199 (1987)
- 5) P. Bertrand, A. Chizzo, T. V. Johnson, M. Shoucri, E. Fijalkov, and M. R. Felix, Phys. Fluids B 2, 1028 (1990)
- 6) T. P. Armstrong, R. C. Harding, G. Knorr, and D. Montgomery, Methods in Computational Physics Vol. 9, B. Alder, S. Fernbach, H. Rotenberg, Eds. Academic Press, New York (1970)
- 7) C. G. Cheng and G. Knorr, J. Comput. Phys. 22, 330 (1976).
- 8) J. Krall, G. Joyce, and Eric Esarey, "Vlasov Simulations of Very-Large-Amplitude Wave Generation in the Plasma Wakefield Accelerator", to appear in Phys. Rev. A (1991)
- 9) H. L. Roulund, P. J. Palmadesso and K. Papadopoulos, Phys. Fluids 24, 832 (1981)

^{*} This work was supported by the U. S. Department of Energy and the Office of Naval Research

Comparison Study: 3D Gyrokinetic PIC Codes on the Cray2, TC2000, and CM2 *

Timothy J. Williams and Y. Matsuda

Lawrence Livermore National Laboratory

1 Introduction

We have ported a 3D electrostatic gyrokinetic PIC code [1] from the Cray-2 to both the DBN TC2000 and the Thinking Machines CM2. On the TC2000, we employ the PFP/PCP split-join parallel programming paradigm [2]. We find reasonably linear scalings for the force computation/particle push, with processor counts up to $O(100)$ (the whole machine). We report on several methods of attack for parallelizing charge accumulation. On the CM2 we employ CM Fortran, coupled with efficient library routines for FFT and scatter operations, the latter being used to parallelize the charge accumulation. On the Cray2, we employ the method of Hiron and Adam [3] to vectorize the charge accumulation and use autotasking to effect some parallelization. We report on our experiences porting the code and show timing comparisons of the code on the three hardware platforms.

2 TC2000

The TC2000 is a MIMD machine with 128 M88100 RISC processors interconnected through a high-bandwidth switch. Each CPU is on a board with 16MB of memory. The switch gives all processors equal access to memory on all other processors boards. Each CPU is rated at 20 MFLOPS (10 MFLOPS for 64 bit arithmetic) yielding a theoretical peak performance of 2 GFLOPS (1 GFLOP for 64-bit arithmetic).

On the TC2000, our basic approach is to use the PCP/PFP extensions to C/Fortran to modify the raw Fortran code to run in parallel; we use interleaved shared memory to store particle and grid arrays. The Parallel C Preprocessor and Parallel Fortran Preprocessor (PCP and PFP) [2][4] are implementations of the split-join parallel programming paradigm. In this paradigm, a fixed-CPU-count team of processors enters the code, all members execute the code unless instructed otherwise (via PCP/PFP embedded statements). The team has a master processor, which can be specifically

*This work performed by LLNL under DoE contract No. W-7405-ENG-48

accessed for scalar code blocks or other special purposes. The team can be split to do independent code blocks or to execute the same code on independent data. Finally, and importantly, all these operations are scalable.

Much of the gyrokinetic PIC code parallelizes in a straightforward manner using the doall parallel do-loop statement in PFP. The force computation and particle push, for example, scale nearly linearly down in execution time with the number of processors in the team. (See Figure 1). The slight sub-linearity is caused by PFP's small static scheduling overhead and contention for shared array elements in the TC2000 switch.

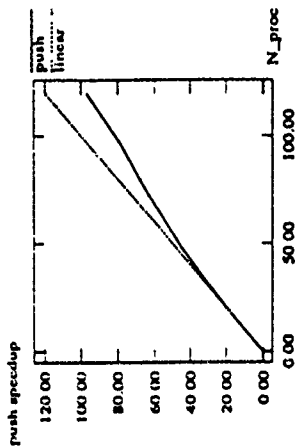


Figure 1: TC2000 speedups for the force computation/particle push compared with ideal (linear) speedup

The charge accumulation is tricky to parallelize and we are investigating several methods to do it. The method used in the timings reported here involves each processor accumulating charge into a local copy of the grid array, then summing all its elements into a global shared grid array (using an array of shared lock variables to avoid data dependencies). Other methods under study include particle sorting using parallel sorts [5] and low-level incremental addition to the charge-density array elements using DBN-specific atomic operations on memory locations.

3 CM2

The Thinking Machines CM2 is a SIMD (Single Instruction Multiple Data) machine with distributed memory. In its full configuration there are 64K bit serial processors connected in a hypercube geometry. In addition, there are 2048 floating-point processors, one for every 32 bit-processors. Each floating-point processor is rated at 16 MFLOPS. Therefore, the theoretical peak performance is 32 GFLOPS.

Many of the benchmarking runs were made on a quadraut (16K processors) of the full configuration. However, timing tests have shown a linear scaling from 16K to 64K processors within several percent. So, we will quote the timings on the full configuration CM2.

The most expensive part of particle codes is usually scatter and gather operations. Scatter operations are required for accumulating the electric charge of particles to grid points every time step, gather operations are required for calculating the force on the particles from the field quantities defined on the grid. To parallelize scatter operations, we employ the utility library routines developed for the CM2 by Thinking Machines Corporation. To parallelize gather operations, we use a "forall" construct of CM Fortran, which is an extension of the Fortran 90 proposed standard.

Calculation of electrostatic potential, which includes the FFT calls, turned out to be a small fraction of the time per step on the CM2 for the number of particles and the size of the grid used in the benchmarking runs reported here. In fact it was negligible when the number of particles per grid cell was greater than about 10. We use the FFT routine in the CMSSL math library.

The timings on the CM2 along with those on the TC2000 and Cray 2 are listed in the table in the last section.

4 Current Results and Conclusions

At the current level of refinement in porting the code to the two parallel machines, we find the results in the following table. All timings are microseconds per particle per timestep. The "jit" annotations refer to the precision of floating point numbers. Runs I and II are for two different parameter sets. Run I has 269001 particles and 16K grid cells; Run II has 1M particles and 256K grid cells. Since the FFT routine is serial on the Cray2 and TC2000 (not vectorized or parallelized), it has a greater effect on execution time than does the parallel FFT on the CM2; thus we show timings with and without inclusion of the time spent in the FFT routines:

	Run I		Run II	
	w/o FFT	w/FFT	w/o FFT	w/FFT
Cray2 (64 bit)	23	27	29	51
TC2000 (32 bit)	20	126	34	613
CM2 32 bit	12	12	19	20
64 bit	16	16	24	26

We have observed a linear scaling in speedup with the number of processors on the CM2. On the TC2000, we only observed near-linear scaling for the particle push/gather; for charge accumulation, it was significantly worse than linear scaling using the current method. We are currently investigating several better parallelization methods; there are still many obvious improvements to be made to the

TC2000 code which will certainly increase overall performance. Discounting the time for the FFT (assuming a vectorized or parallel FFT will be negligible in typical runs, which have more than 10 particles per cell) we find from the table that the code takes 16μsec, 40μsec, and 23μsec per particle per step on the CM2, TC2000, and Cray2, respectively, with 64 bit arithmetic (the time for the TC2000 was multiplied by 2 as a upper-bound estimate of the penalty for converting to 64 bits).

References

- [1] Developed by R. D. Sydora, W. W. Lee, and others.
- [2] Eugene D. Brooks III, UCRL-99073, LLNL (1988).
- [3] A. Héron and J. C. Adam, J. Comput. Phys. 85, 284 (1989).
- [4] FFT authors: E. D. Brooks III and K. Warren, LLNL.
- [5] T. J. Williams, Y. Matsuda, and E. Boerner, in UCRL-ID-107522.

A Multiple Species, Variable Weight Extension of the

Zero-Electron-Mass Hybrid Model

D W Hewett, G. F. Simonson, and J. A. Byers
University of California
Lawrence Livermore National Laboratory
Livermore, California 94550

The Basic Zero-Electron-Mass Hybrid Model

The zero-electron-mass hybrid simulation model describes microscopic phenomena while including particle ion effects.^{1,2,3,4} Working in the quasineutral limit and neglecting electron inertia, Hewett's 2.5D RZ version of this model² combines the electron momentum equation

$$\mathbf{E} = -\frac{\nabla n T_e}{en} - \frac{u_e}{c} \times \mathbf{B} + \eta \mathbf{J} \quad (1)$$

with radiation-free or Darwin limit of Maxwell's equations

$$\nabla \times \mathbf{B} = \frac{4\pi}{c} \mathbf{J} \quad (2)$$

$$\nabla \times \mathbf{E} = -\frac{1}{c} \frac{\partial \mathbf{B}}{\partial t} \quad (3)$$

to obtain, in the Coulomb gauge, the strongly coupled set of equations

$$A_\theta - \frac{c^2}{4\pi} \nabla^2 A_\theta + u_{e,r} \frac{1}{r} \frac{\partial(r A_\theta)}{\partial r} + u_{e,z} \frac{\partial A_\theta}{\partial z} = 0 \quad (4)$$

$$\begin{aligned} B_\theta - \frac{c^2}{4\pi} \left[\frac{\partial}{\partial r} \left(\frac{\eta}{r} \frac{\partial(r B_\theta)}{\partial r} \right) + \frac{\partial}{\partial z} \left(\eta \frac{\partial B_\theta}{\partial z} \right) \right] + \frac{\partial}{\partial r} (u_{e,r} B_\theta) + \frac{\partial}{\partial z} (u_{e,z} B_\theta) \\ = B_r \left(\frac{\partial u_{e,\theta}}{\partial r} - \frac{u_{e,\theta}}{r} \right) + B_z \frac{\partial u_{e,\theta}}{\partial z} - \frac{c}{en} \left[\frac{\partial}{\partial z} \left(\frac{\partial n}{\partial r} \frac{\partial T_e}{\partial r} \right) - \frac{\partial n}{\partial r} \frac{\partial T_e}{\partial z} \right] \end{aligned} \quad (5)$$

for the time dependence of the normal (here θ) components of the magnetic vector potential A_θ (hence, B_r and B_z) and the magnetic field B_θ . The electron flow velocity components $u_{e,r}$ and $u_{e,z}$ come from the curl of $B_\theta \hat{e}_\theta$, using

$$\mathbf{u}_e = \mathbf{u}_r - \frac{c}{4\pi en} \nabla \times \mathbf{B} \quad (6)$$

where the density and ion flow velocities are n , $u_{i,r}$, and $u_{i,z}$, respectively, and are obtained from the ion PIC advance

After the B field has been advanced in time, the total E field in finite density cells is now calculated from the basic equation for E eqn. (1) using the newly updated B field and u_e . The field calculations are completed by solving $\nabla^2 E = 0$ in the vacuum region. To complete this model, a mechanism for advancing T_e was developed that allows rapid thermal equilibration along field lines. This algorithm was discussed in reference (5).

Solutions in the Low and Zero Density Regions

A common difficulty with codes that use the momentum equation to construct E is the problem of low density fluctuations that arises from the required division by density. Clearly a different set of equations must be invoked when the density is zero, but even in regions of finite density, a low density fluctuation generates a spike in the local Alfvén signal velocity. If this velocity spike exceeds the local Courant stability limit, it generally causes an abrupt termination of an explicit simulation. The numerical origin of this difficulty is easily seen. Using eqn. (6) to eliminate u_e in eqn. (1), taking the curl of the resulting eqn (1), and substituting into Faraday's law eqn. (3) to get an equation for B leads to an expression in which one of the terms looks like

$$\dot{B} \sim \frac{B}{n} \nabla^2 B. \quad (7)$$

Standard explicit finite differencing in time and space gives a numerical constant for stability relating the time step Δt to the spatial grid resolution Δx of the form

$$\Delta t \leq \frac{n}{B} (\Delta x)^2 \quad (8)$$

that clearly displays the numerical issues that arise when dealing with the low density regions. It is obvious that good spatial resolution coupled with low densities will place severe restrictions on the time step—especially since typical plasmas have the largest B fields where the density is smallest. Further, PIC representations by their nature are subject to very high fluctuations in density, and $\delta n/n$ is largest where n is smallest.

The problem is traditionally handled by declaring a density cutoff. When the computational cell density falls below this cutoff, our model considers that cell to be a vacuum and the basic equations are replaced by equations representing Maxwell's equations in the limit of instantaneous propagation. This abrupt change in the character of the governing equations and the associated signal propagation speed are primarily responsible for the "sensitivity" algorithms employing explicit field solutions display around these vacuum-plasma interfaces. When any ion species is represented by PIC, these interface locations tend to move or flicker around stochastically from one time step to the next—especially in regions with relatively few particles per cell. These issues force explicit hybrid code users to be very conservative with their time step selection, to use larger than necessary numbers of particles in their PIC representations, and to use larger than necessary cutoff values.

An implicit field solution greatly reduces these difficulties; the trade off is that the algorithm is not easy to implement. Since the explicit PIC technique will impose a time step constraint on the total algorithm, the electron-field algorithm need not be "fully implicit" to make the file algorithm the most robust part of the code. Our ADI procedure handles the disparity in signal velocities between adjacent plasma and vacuum cells without instability. In addition, the method has the capability to handle the occurrence of vacuum regions whether true or single "vacuum" cells stochastically imbedded in finite density regions because of the reduced need to maintain small, smooth time derivatives. The vacuum solution can be generated simultaneously with the plasma solution by simply setting the vacuum relativity to a "large" value so that all terms other than $\nabla^2 A_\theta$ and

V^2B_e become insignificant. This technique removes the need for vacuum-plasma interface monitoring and, as a result, the B field advance can function with boundary conditions applied only on the boundaries of the simulation region with no restrictions on the position or number of plasma-vacuum interfaces. The actual implementation is accomplished with noniterative ADI. Surprisingly good performance has been obtained with only marginal numbers of particles. The algorithm is "straightforward" but tedious; the gory details are contained in reference (2).

An alternative procedure sometimes used in quasi-neutral Darwin simulation codes is to create an artificial density minimum in place of the actual vacuum region. This procedure thus prevents dividing by too small a number, and the hope is that with a reasonably small density minimum, not much damage to the desired physics will occur. However, a quasineutral algorithm absolutely requires the interpretation that a low density region of electrons be balanced by an equal density of ions. The trouble is that there are no simulation particles representing these low density region ions, so no ion currents are calculated, despite the presence of electromagnetic fields that can drive them. The resulting conclusion is that the charge-canceling ions in this low density region really must be infinitely massive. This creates a region with zero Alfvén speed (despite the low density used) that is exactly the opposite of the large or infinite Alfvén velocity desired for a vacuum region. A density floor may be an attractive, easy solution to dealing with the plasma-vacuum interface, but at the least it causes very troublesome interpretation problems; at the worst it may produce the wrong results by imposing exactly the wrong type of boundary conditions on the solution in the plasma region.

Generalizing to Multiple Species

Multiple species generalizations of the PIC algorithm are easy; we declare the next group of particles to be a new species with a new charge and mass and gather the source terms in new arrays. The electron/field representation requires a little more thought. In a one-ion-species quasi-neutral algorithm the electron density is assumed to be indistinguishable from the ion density; in all cases where an electron density is required, we substitute the ion density. Our technique for generalization of this algorithm to multiple species is to maintain the separate identity of each species until the start of the field advance. At the start of the field advance we construct a new set of equivalent proton source terms. These quantities are generated in the obvious way: we sum the electrical charge and current densities over species in each cell and then form the proton source terms by dividing by the charge of a proton. Our algorithm makes explicit use of the ion (now proton) drift velocity, so we divide the total electric current by the total electric charge density in each cell to generate the needed proton drift velocity.

It is tempting to incorporate smoothing in the electric charge density before the division but we have been rather painfully convinced that such averaging can cause the algorithm to fail some rather elementary momentum conservation tests in some applications.

Variable Weight Extension

In the problem of interest to us now, magnetic debris containment⁶, we study extreme

ranges of cylindrical radius. Populating computational zones of grossly different volumes with identical simulation particles requires an exorbitant number of particles to place even a few into the smaller volumes represented by our uniform radial cells near the mesh symmetry axis. Since in this problem we do not expect very much mixing of the background material on the global scale, we can improve the particle statistics in these inside zones and minimize the total number of particles required. We have implemented a variable number of real-particles-per-simulation particle that begins with a preselected number of simulation particles per cell, and then uses the desired initial density to determine the number of real particles represented by every simulation particle initialized within that cell. Each particle carries this fixed sixth descriptor throughout the calculation (in addition to its five changing components for velocity and position). The weighting is applied every time particles are collected on the mesh, to calculate true densities, velocities, etc. This scheme has the additional advantage that relatively stable initial density profiles spanning broad ranges can be described with variable weighting while maintaining a more nearly constant number of particles per cell.

Acknowledgements

Work performed under the auspices of the United States Department of Energy by the Lawrence Livermore National Laboratory under contract number W-7405-ENG-48

REFERENCES

1. J. A. Dyers, B. I. Cohen, W. C. Condit, and J. D. Hanson, *J. Comp. Phys.* **27** (1978) 363.
2. D. W. Hewett, *J. Comp. Phys.* **38** (1980) 378.
3. D. S. Harned, *J. Comp. Phys.* **47** (1982) 452.
4. A. Mankofsky, R. N. Sudan, and J. Denavit, *J. Comp. Phys.* **70** (1987) 89.
5. D. W. Hewett, *Nucl. Fusion* **24** (1984) 349.
6. G.F. Simonson and D.W. Hewett, *Recent Magnetic Debris Containment Calculations*, UCRL-ID-107411, April, 1991.

A 2.5D RZ Streamlined Darwin Field Code and its Application to Mirrortron Simulations

D J Larson and D W Hewett
University of California
Lawrence Livermore National Laboratory
Livermore, California 94550

Plasma simulations which use the Darwin approximation of Maxwell's equations have been limited, by boundary condition difficulties, to those cases in which the plasma is "far away" from all boundaries. The Streamlined Darwin Field (SDF) equations¹ provide an alternative solution path in which the need to specify the most troubling boundary conditions is eliminated.

We use a novel solution technique² to solve the strongly coupled set of partial differential equations arising from the SDF equations and have implemented this new Darwin procedure in the PIC code BEAGLE. The code is being used to study the Mirrortron³, a device for generating large transient potentials within a mirror confined plasma.

Streamlined Darwin Field Equations

The traditional Darwin model has been motivated and discussed by several authors. The work of Nielson and Lewis⁴ and the streamlined version of Hewett and Boyd¹ are particularly relevant to the brief details and summary we give here. We start with the Darwin limit of Maxwell's equations, in which the solenoidal part of the displacement current has been discarded.

$$\nabla \times \mathbf{E} = -\frac{1}{c} \frac{\partial \mathbf{B}}{\partial t}$$

$$\nabla \times \mathbf{B} = \frac{4\pi}{c} \mathbf{J} + \frac{1}{c} \frac{\partial \mathbf{E}_{\text{sol}}}{\partial t} = \frac{4\pi}{c} \mathbf{J}_{\text{sol}} \quad (1)$$

Introducing the solenoidal vector potential \mathbf{A} , such that $\mathbf{B} = \nabla \times \mathbf{A}$ and $\nabla \cdot \mathbf{A} = 0$, we have

$$\nabla^2 \mathbf{A} = -\frac{4\pi}{c} \mathbf{J}_{\text{sol}}, \quad (2)$$

and the equation for the electrostatic potential ϕ is the usual

$$\nabla^2 \phi = -4\pi \rho. \quad (3)$$

Using $\mathbf{E} = -\nabla \phi - \frac{1}{c} \frac{\partial \mathbf{A}}{\partial t}$, we obtain the irrotational part of the electric field from the gradient of ϕ . The equation for \mathbf{E}_{sol} comes from the time derivative of eqn. (2) with the above definition of \mathbf{E} -resulting in

$$\nabla^2 \mathbf{E}_{\text{sol}} = \frac{4\pi}{c} \mathbf{J}_{\text{sol}}. \quad (4)$$

As in Nielson and Lewis,

$$\frac{4\pi}{c} \mathbf{j} = \mathbf{K} + \zeta \times \mathbf{B} + \mu(\mathbf{E}_{\text{irr}} + \mathbf{E}_{\text{sol}}). \quad (5a)$$

We define \mathbf{Q} such that

$$\frac{4\pi}{c^2} \mathbf{j} = \mathbf{Q} + \mu \mathbf{E}_{\text{sol}}, \quad (5b)$$

so that eqn. (5) becomes

$$\nabla^2 \mathbf{E}_{\text{sol}} = [\mathbf{Q} + \mu \mathbf{E}_{\text{sol}}]_{\text{sol}}. \quad (6)$$

These equations comprise the Darwin limit of Maxwell's equations. For many applications the major stumbling block is the decomposition of \mathbf{j} , the quantity in the brackets, because it requires the specification of physically obscure boundary conditions. Fortunately, the SDF procedure eliminates the need for this decomposition. The SDF equations are arrived at by the following manipulations, starting with eqn. (4):

$$\nabla^2 \mathbf{E}_{\text{sol}} = \frac{4\pi}{c^2} (\mathbf{j} - \mathbf{j}_{\text{irr}}) \quad (7)$$

$$\nabla^2 \mathbf{E}_{\text{sol}} + \frac{4\pi}{c^2} \mathbf{j}_{\text{irr}} = \frac{4\pi}{c^2} \mathbf{j} \quad (8)$$

Defining

$$-\nabla^2(\nabla\psi) \equiv \frac{4\pi}{c^2} \mathbf{j}_{\text{irr}} \quad (9)$$

we have

$$\nabla^2(\mathbf{E}_{\text{sol}} - \nabla\psi) = \frac{4\pi}{c^2} \mathbf{j}. \quad (10)$$

Comparing with eqn. (4), eqn. (10) contains a new term on the left hand side that just compensates for the irrotational term that we have retained on the right. Defining

$$\xi \equiv \mathbf{E}_{\text{sol}} - \nabla\psi \quad (11)$$

we have

$$\nabla^2 \xi = \frac{4\pi}{c^2} \mathbf{j} \quad (12)$$

Using eqn. (5b) we may further modify eqn. (12) as

$$\nabla^2 \xi = \mathbf{Q} + \mu \mathbf{E}_{\text{sol}}$$

$$\nabla^2 \xi - \mu \xi = \mathbf{Q} + \mu \nabla\psi.$$

From the definition, eqn. (11), we note that

$$\xi_{nr} = -\nabla\psi$$

so that

$$\nabla^2\psi = -\nabla\cdot\xi.$$

The equation for E_{tot} , eqn (4), then becomes the two coupled equations

$$\nabla^2\xi - \mu\xi = Q + \mu\nabla\psi \quad (13)$$

$$\nabla^2\psi = -\nabla\cdot\xi \quad (14)$$

These equations are coupled by the parameter μ , which is the square of the inverse of the collisionless electromagnetic skin depth.

SDF Solution Technique

We are interested in solutions for regimes in which μ is large compared to other terms in the equations. This means the equations are strongly coupled and solution by Picard iteration is slow and often diverges. We apply an extension of the dynamic alternating direction implicit (DADI) algorithm of Doss and Miller⁵ in order to find a simultaneous solution of these equations. Letting H (V) represent the horizontal (vertical) part of the finite-differenced ∇^2 operator, our splitting of the coupled equations above is:

H - PASS

$$(-\omega + H - \mu/2)\xi_r^{n+1} = (-\omega - V' + \mu/2)\xi_r^n + \mu D_x\psi^n + Q_x$$

$$(-\omega + H - \mu/2)\xi_v^{n+1} = (-\omega - V + \mu/2)\xi_v^n + \mu D_y\psi^n + Q_y$$

$$(-\omega + H)\psi^{n+1} = (-\omega - V)\psi^n - D_x\xi_r^n - D_y\xi_v^n \quad (15a)$$

V - PASS

$$(-\omega + V - \mu/2)\xi_r^{n+1} = (-\omega - H + \mu/2)\xi_r^n + \mu D_x\psi^n + Q_x$$

$$(-\omega + V - \mu/2)\xi_v^{n+1} = (-\omega - H + \mu/2)\xi_v^n + \mu D_y\psi^n + Q_y$$

$$(-\omega + V)\psi^{n+1} = (-\omega - H)\psi^n - D_x\xi_r^n - D_y\xi_v^n \quad (15b)$$

As in simple ADI, an iterated solution is obtained by solving the equations in the horizontal (H) pass, using the latest values for the second-order unknowns on the right side, followed by the vertical (V) pass, using the results from the previous H pass and so on until convergence.

This algorithm was tested in cartesian and cylindrical coordinates and compared for the cartesian version, with a biconjugate gradient solver.⁶ Our results show that DADI is a factor of 4 to several orders of magnitude faster than Anderson's BCG algorithm which we used as a "black box"

Mirrortron Simulations

We have incorporated the new Darwin field procedure in the RZ simulation code BEAGLE, an extension of GYMNOS.⁷ The first application of the new code will be to the Mirrortron.³ The Mirrortron is a device for generating large transient potentials within a mirror confined plasma. The transient potential is generated by the application of an additional local mirror field produced by a strap coil. The rise time of the current in the strap coil is on the order of tens of nanoseconds. This allows the hot electrons to respond to the local mirror force while the ions remain inertially confined. The potentials generated by this method are predicted, and have now been confirmed by simulation, to have accelerating and focusing contours. We plan to compare Mirrortron behavior with and without inductive fields, and evaluate the Mirrortron's ability to accelerate heavy ion beams.

Acknowledgments

Work performed under the auspices of the United States Department of Energy by the Lawrence Livermore National Laboratory under contract number W-7405-ENG-48.

References

1. D.W. Hewett and J.K. Boyd, *J. Comp. Phys.* 70 (1987), 166.
2. D.W. Hewett, D.J. Larson, S. Doss, *J. Comp. Phys.* (accepted) 1991. (1987), 878
3. R.F. Post, *Phys. Rev. Lett* 58 (1987), 878.
4. C.W. Nielson and H.R. Lewis, *Meth. Comput. Phys.* 10 367, B. Alder, S. Fernbach, M. Rotenberg, and J. Killeen, eds., Academic Press, New York, 1976
5. S. Doss and K. Miller, *SIAM J. Numer. Anal.* 10 (1979), 837.
6. D.V. Anderson, A.E. Koniges, and D.E. Shumaker, *Comput. Phys. Commun.* 51 (1988), 391.
7. D.W. Hewett and D.J. Larson, *The Best of GYMNOS: A User's Guide UCRL Internal Report*, (in preparation, 1991).

ONE DIMENSIONAL HYBRID CODE INCLUDING ELECTRON INERTIAL EFFECTS .

E.SAUSSEDE

Université de BORDEAUX I (ADERA BP N°48 St.Médard-en-Jalles, FRANCE)

The interaction between a very dense hot plasma moving rapidly within a less dense warm magnetized plasma is analyzed. Three different approaches can be used for plasma simulation :

- one using many particle models in order to obtain detailed kinetic behavior .
- one using fluid models which consist in solving numerically the MHD equations and the need for a kinetic treatment of the ions to reproduce the correct ions dynamics and the corresponding shock structure is essential .

We present a 1D spatial 3D vector field components hybrid simulation code where :

- ions are supposed to be collisionless macroparticles .
 - electrons are treated as fluid with inertial effects retained .
 - electromagnetic field is computed selfconsistently solving Maxwell equations .
- Thus, the system of equations derives from Maxwell equations, equation of motion for ions, particles and electrons fluid conservation laws (the three first moments of Vlasov equation) .

The computational cycle follows as :

Given electromagnetic field components on a grid, we interpolate them to the particles, the particles are moved according to the forces exerted on them by these fields and then the charge and current densities are interpolated to the grid . To end the cycle, the fields are solved on the grid consistently with the charge and current densities to prepare for the next cycle .

ADVANCING THE PARTICLES

Once the fields are known at the particle positions, the particles are advanced from their old position to their new position by integrating the equation of motion .

$$M_i \frac{d\vec{V}_i}{dt} = q_i (\vec{E} + \vec{V}_i \times \vec{B} - \eta \vec{j})$$

with η : phenomenological resistivity .

M_i : ion mass .

q_i : ion charge .

We integrate these equations using a leap-frog scheme .

SOLVING THE FIELD EQUATIONS .

Once, the particle push gave us the charge and current ions densities at an advanced time step, the electromagnetic field is computed using a Newmark time implicit algorithm.

Maxwell system writes :

$$\begin{cases} \frac{\partial \vec{B}}{\partial t} = -\nabla \times \vec{E} \\ \epsilon_0 \frac{\partial \vec{E}}{\partial t} = \frac{1}{\mu_0} \nabla \times \vec{B} - \vec{j} \end{cases}$$

We can rewrite it as :

$$\begin{cases} \frac{\partial^2 B_x}{\partial t^2} - c^2 \frac{\partial^2 B_y}{\partial x^2} = \frac{1}{\epsilon_0} \frac{\partial j_z}{\partial x} \\ \frac{\partial^2 B_x}{\partial t^2} - c^2 \frac{\partial^2 B_z}{\partial x^2} = \frac{1}{\epsilon_0} \frac{\partial j_y}{\partial x} \end{cases}$$

provided a finite difference discretization in space we obtain the following system :

$$\begin{cases} M \frac{\partial^2 B_y(i)}{\partial t^2} - K B_y(i) = S_y(i) \\ M \frac{\partial^2 B_z(i)}{\partial t^2} - K B_z(i) = S_z(i) \end{cases}$$

where :

$$M = Id$$

$$K = -\frac{c^2}{\Delta x} \begin{vmatrix} -2 & 1 & 0 & 0 \\ 1 & -2 & \dots & 0 \\ 0 & \dots & \dots & 1 \\ 0 & 0 & 1 & -2 \end{vmatrix}$$

Newmark's method uses a Taylor series expansion to construct linear relationships of the following type which are limited to second derivatives .

$$(M + \beta \frac{\Delta t^2}{2} K) B^{n+1} = (2M - (\frac{\Delta t^2}{2} \beta + \gamma) \Delta t^2 K) B^n + (-M + (\gamma - \frac{1}{2} \beta) \Delta t^2 K + \Delta t^2 S^n$$

with :

$$\beta = \gamma = \frac{1}{2}$$

which makes the scheme second order accurate and unconditionally stable .

SOLVING ELECTRONS FLUID COMPONENTS .

Equations on conservative form write :

$$\left\{ \begin{array}{l} \frac{\partial \rho_e}{\partial t} + \frac{\partial (\rho_e V_x)}{\partial x} = 0 \\ \frac{\partial \rho_e V_x}{\partial t} + \frac{\partial (\rho_e V_x^2 + P_e)}{\partial x} = S_1 \\ \frac{\partial \rho_e V_y}{\partial t} + \frac{\partial (\rho_e V_x V_y)}{\partial x} = S_2 \\ \frac{\partial \rho_e V_z}{\partial t} + \frac{\partial (\rho_e V_x V_z)}{\partial x} = S_3 \\ \frac{\partial \mathcal{E}_e}{\partial t} + \frac{\partial (\mathcal{E}_e + P_e) V_x}{\partial x} = S_4 \end{array} \right.$$

with :

$$\left\{ \begin{array}{l} S_1 = q_e n_e (E_x + V_y B_z - V_z B_y - q_e n_e \eta J_x) \\ S_2 = q_e n_e (E_y + V_z B_x - V_x B_z - q_e n_e \eta J_y) \\ S_3 = q_e n_e (E_z + V_x B_y - V_y B_x - q_e n_e \eta J_z) \\ S_4 = \eta J^2 + V_x S_1 + V_y S_2 + V_z S_3 \end{array} \right.$$

where :

m_e : electron mass
 q_e : electron charge .

and : $\rho_e = m_e \cdot n_e$.

The previous system is solved provided a Van Leer splitting corrected by a Flux Corrected Transport method (FCT) .

REFERENCES

- C.K.BIRDSALL, A.B.LANGDON : "PLASMA PHYSICS VIA COMPUTER SIMULATION". Mac GRAW-HILL company (1985).
- R.W HOCKNEY,J.W EASTWOOD : "COMPUTER SIMULATION USING PARTICLES".Adam Hilger, Bristol and Philadelphia (1988).
- F.F CHIEN : "INTRODUCTION TO PLASMA PHYSICS AND CONTROLLED FUSION".Plenum Press , 2nd edition (1984).
- B. VAN LEER : "FLUX VECTOR SPLITTING FOR EULER EQUATIONS".Lecture notes in physics 170, 507 (1982) .
- J.P.BORIS & D.L.BOOK.: "LSHASTA, A FLUID TRANSPORT ALGORITHM THAT WORKS".J.C.P. 11, 38-69, (1973) .
- E.J. HOROWITZ: "ON3D : A 3D QUASI-NEUTRAL HYBRID PIC CODE WITH APPLICATIONS TO THE TILT MODE INSTABILITY IN FIELD REVERSED CONFIGURATIONS".Ph.D Thesis (1987) .
- D.WINSKE & M.ERROY: "HYBRID SIMULATION TECH APPLIED TO THE EARTH'S BOW SHOCK".Lecture at ISSSI, RIDEL Holland, (1984) .

An Implicit Hybrid, Quasi-Neutral Model For Low Frequency, Electromagnetic Plasma Simulation

II X. Vu and J. U. Brackbill
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

An implicit, quasi-neutral hybrid model is presented in which all species but one are treated kinetically using finite-size particles. The single species which is not treated using finite-size particles is modeled as a fluid for which charge and current densities and internal energy evolve corresponding to the usual conservation equations (mass, momentum, and energy). The energy equation can also be replaced by an equation of state, e.g., isothermal or adiabatic process, depending on the particular problems one tries to model. A one-dimensional version of this implicit, quasi-neutral hybrid model has been implemented, and a comparison of the simulation results using this code against those obtained with existing one-dimensional explicit hybrid codes is underway. The result of this comparison will be presented in detail at the time of presentation.

1. INTRODUCTION

The principal goal of plasma simulations using finite-size particles is to describe correctly the kinetic behavior of the individual constituents of the plasmas, and their effects on the plasmas' macroscopic behavior. For a large number of physical problems, in order to obtain the correct macroscopic plasma behavior, it is necessary to include the kinetic behavior of all parts of the plasma, and there exists a number of numerical algorithms for this purpose (cf. [1-16]). However, it is costly to treat all parts of the plasma kinetically using finite-size particles. There are physical problems for which the electrons, the lightest constituent of the plasma, do not contribute to the macroscopic plasma behavior, i.e., the electrons do nothing other than providing a background charge density to ensure that the plasma is charge-neutral (cf. [17]). For this particular class of problems, it is more economical, without loss of physics, to treat the electrons as a fluid whose macroscopic properties are described by a set of conservation equations. A number of hybrid (particle fluid) algorithms are available for this class of physical problems. These hybrid algorithms are variations of one central theme: the electrons are modeled as a fluid whose motion provides local charge neutrality for the plasma, i.e., $\nabla \cdot \mathbf{J} = 0$. One class of hybrid algorithms totally neglects the electron inertial effect (cf. [18]), and thus completely removes the shortest time scales, i.e., the electron plasma and cyclotron periods. There are physical problems where although the electron kinetic effect does not play an important role, its finite-mass effect is necessary to produce the correct physical results. For example, for problems with $\mathbf{k} \cdot \mathbf{B} \approx 0$, it might be necessary to include the electron finite-mass effect since the electrons are constrained to the field lines, and therefore cannot move freely across the field lines to provide instantaneous charge neutrality. Another class of hybrid algorithms improves this shortcoming by partially including the electron-finite mass effect in the transverse direction, but the electron finite-mass effect in the longitudinal direction is completely ignored (cf. [19]). Recently, an implicit hybrid model is proposed in which the electron finite-mass effect in both the transverse and longitudinal directions are partially included by retaining the $\partial \mathbf{J}_\parallel / \partial t$ term in the electron momentum equation (cf. [20]). However, the $\nabla \cdot (\mathbf{J}_\perp \mathbf{J}_\perp / \rho_e)$ term in the electron momentum equation is completely neglected. Here, we propose a hybrid model in which the full electron finite-mass effect is retained. A one-dimensional version of this model has been implemented, and application of this code to the study of perpendicular fast shocks and oblique switch-off shocks is ongoing. Preliminary results indicate that the code yields the correct shock speeds for, and the correct Rankine-Hugoniot conditions for, both types of shocks. The features of the shocks, and a comparison of the results with those obtained with: (1) a one-dimensional hybrid code where the electron finite-mass effect

is completely neglected [17], and (2) an implicit, full-particle code (CELESTID) [16] where the electrons are treated kinetically using finite-size particles, will be presented in detail at the time of presentation.

II. FORMULATION

The motion of an ion in an electromagnetic field is described by:

$$\left. \begin{aligned} \frac{d\mathbf{x}}{dt} &= \mathbf{u} \\ \frac{d\mathbf{u}}{dt} &= \frac{q}{m} \left(\mathbf{E} + \frac{\mathbf{u} \times \mathbf{B}}{c} \right) \end{aligned} \right\} \quad (1)$$

The electromagnetic field influencing the ion motion is described by Maxwell's equations. For convenience, the \mathbf{E} and \mathbf{B} fields are formulated in terms of the scalar and vector potentials using the Coulomb gauge:

$$\left. \begin{aligned} \frac{1}{c^2} \frac{\partial^2 A}{\partial t^2} - \nabla^2 A &= \frac{4\pi}{c} \mathbf{J} - \frac{1}{c} \frac{\partial \nabla \phi}{\partial t} \\ \nabla^2 \phi &= -4\pi \rho \\ \mathbf{B} &= \mathbf{b} + \nabla \times \mathbf{A} \\ \mathbf{E} &= -\nabla \phi - \frac{1}{c} \frac{\partial \mathbf{A}}{\partial t} \end{aligned} \right\} \quad (2)$$

where $\nabla \cdot \mathbf{A} = 0$, \mathbf{b} is the external magnetostatic, uniform field. The electron current evolves corresponding to the equation for momentum conservation,

$$\frac{\partial \mathbf{J}_e}{\partial t} + \nabla \cdot \left(\frac{\mathbf{J}_e \mathbf{J}_e}{\rho_e} \right) = \frac{q_e}{m_e} \left(\rho_e \mathbf{E} + \frac{\mathbf{J}_e \times \mathbf{B}}{c} - \nabla p_e \right) \quad (3)$$

where $p_e = (\rho_e / q_e) T_e$. The charge neutrality condition is ensured by the following conditions:

$$\left. \begin{aligned} \rho_e &= -\rho_i \\ \nabla \cdot \mathbf{J} &= 0 \end{aligned} \right\} \quad (4)$$

An electron energy equation can be added to describe the electron energy transport process. For our current physical applications, an electron energy equation is not used. Instead, the electrons are modeled as either an isothermal fluid, $T_e = T_{e0}$, or an adiabatic fluid, $T_e = T_{e0}(\rho_e / \rho_{e0})^\gamma$, with $\gamma_e = 5/3$.

Using the time-implicit algorithm, Eq. (3) is discretized as follows:

$$\frac{\mathbf{J}_{e1} - \mathbf{J}_{e0}}{\delta t} + \nabla \cdot \left(\frac{\mathbf{J}_{e0} \mathbf{J}_{e0}}{\rho_{e0}} \right) = \frac{q_e}{m_e} \left(\rho_{e0} \mathbf{E}_0 + \frac{\mathbf{J}_{e0} \times \mathbf{B}_0}{c} - \nabla p_{e0} \right) \quad (5)$$

where $\mathbf{J}_e = (\mathbf{J}_{e1} + \mathbf{J}_{e0})/2$. Solving the above equation for \mathbf{J}_{e1} , one obtains:

$$\mathbf{J}_{e1} = \mathbf{J}_{e0} - \frac{1}{4\pi} \left(\frac{\delta t}{\rho_{e0}} \right) \mu_e \cdot \mathbf{E}_0$$

with

$$\mathbf{J}_{e0} = -\left(\frac{1}{4\pi} \right) \mu_e \cdot \left[\frac{1}{m_e \rho_{e0}} \left[\mathbf{J}_{e0} - \frac{\delta t}{2} \nabla \cdot \left(\frac{\mathbf{J}_{e0} \mathbf{J}_{e0}}{\rho_{e0}} \right) \right] - \frac{\delta t}{2} \left(\frac{\nabla p_{e0}}{\rho_{e0}} \right) \right] \quad (6)$$

$$\mu_e(x) = \frac{1\pi q_e q_i / m_e}{1 + \left[\frac{q_e q_i(x) \delta t}{m_e c} \right]^2} \left[1 - 1 \times \left(\frac{q_e B_0(x) \delta t}{m_e c} + \left(\frac{q_e B_0(x) \delta t}{m_e c} \right)^2 \left(\frac{q_e B_0(x) \delta t}{m_e c} \right)^2 \right) \right] \quad (7)$$

is the unit tensor. Following the approach of the implicit moment method [16], the time centered ion current can be written in the same form:

$$J_i = J_{i0} - \frac{1}{4\pi} \left(\frac{\delta t}{2} \right) \mu_i \cdot E_0 \quad (8)$$

where J_{i0} consists of the contributions from the ion current and pressure gradient (i.e. [16] for more detail). The tensor μ_i is defined in the same way as μ_e , and is summed over all ion species. Because of the assumption that the plasma is quasi neutral, i.e. $\rho_e \phi = -\rho_i \phi$, one can no longer use the Poisson's equation to determine the electric field. Instead, the electric field is obtained by requiring that $\nabla \cdot (J_e + J_i) = 0$ which, when used in conjunction with Eqs (5) and (8), yields:

$$\frac{\delta t}{2} \nabla \cdot (\mu_e + \mu_i) \cdot E_0 = -1\pi \nabla \cdot (J_i + J_e) \quad (9)$$

Eq. (9), rather than the Poisson's equation, is used to determine the electric field. The electric field obtained from Eq. (9) can be thought of as the electric field necessary to keep the plasma quasi neutral instead of one caused by local charge imbalance. The magnetic field is obtained by solving for a discretized version of Eq. (2)

$$\left. \begin{aligned} \frac{1}{c^2} \frac{\partial A_1}{\partial t^2} - \frac{2A_0 + A_1}{\delta t^2} - \nabla^2 A_0 &= \frac{4\pi}{c} (J_{i1} + J_{e1}) - \frac{1}{c} \left(\frac{\delta t}{2} \right) (\mu_e + \mu_i) \cdot E_0 \\ &\quad - \frac{1}{c} \frac{\nabla \phi_0 - \nabla \phi_0}{\delta t} \\ A_0 &= \theta A_1 + (1 - \theta) A_0 \\ \phi_0 &= 0 \\ E_0 &= -\nabla \phi_0 - \frac{1}{c} \frac{A_1 - A_0}{\delta t} \\ B_1 &= B + \nabla \times A_1 \end{aligned} \right\} \quad (10)$$

Eqs. (9) (10) are solved for the time advanced scalar and vector potentials (ϕ_0 and A_1), and thus provide a complete description of the time advanced self-consistent electromagnetic fields.

REFERENCES

1. D. Dickman, R. I. Morse and C. W. Nielson, *Phys. Fluids* **12**, 1708 (1969).
2. A. B. Langdon, *J. Comput. Phys.* **6**, 247 (1970).
3. A. B. Langdon and B. F. Lasinski, *Methods Comput. Phys.* **16**, 327 (1976).
4. C. W. Nielson and R. I. Morse, *Methods Comput. Phys.* **16**, 367 (1976).
5. J. Riccardo Neto, P. L. Pritchett, A. T. Lin and J. M. Dawson, *J. Comput. Phys.* **23**, 369 (1977).
6. D. W. Hewett, *J. Comput. Phys.* **38**, 378 (1980).
7. R. J. Mason, *J. Comput. Phys.* **41**, 233 (1981).
8. J. Debnat, *J. Comput. Phys.* **42**, 337 (1981).
9. J. U. Brackbill and D. W. Forslund, *J. Comput. Phys.* **46**, 271 (1982).
10. D. W. Forslund and J. U. Brackbill, *Phys. Rev. Lett.* **48**, 1614 (1982).

11. J. M. Dawson, *Rev. Mod. Phys.* **55**, 403 (1983).
12. R. J. Mason, in *Multiple Time Scales*, edited by J. U. Brackbill and B. I. Cohen (Academic Press, New York, 1985), Chap. 8.
13. J. U. Brackbill and D. W. Forslund, in *Multiple Time Scales*, edited by J. U. Brackbill and B. I. Cohen (Academic Press, New York, 1985), Chap. 9.
14. B. I. Cohen, in *Multiple Time Scales*, edited by J. U. Brackbill and B. I. Cohen (Academic Press, New York, 1985), Chap. 10.
15. A. B. Langdon and D. C. Barnes, in *Multiple Time Scales*, edited by J. U. Brackbill and B. I. Cohen (Academic Press, New York, 1985), Chap. 11.
16. H. X. Vu and J. U. Brackbill, *Comput. Phys. Commun.*, submitted for publication.
17. D. Winske and M. M. Leroy, *J. Geophys. Res.* **89**, 2673 (1981).
18. J. A. Byers, B. I. Cohen, W. C. Condit, and J. D. Hanson, *J. Comput. Phys.* **27**, 363 (1978).
19. D. W. Hewett and C. W. Nielson, *J. Comput. Phys.* **29**, 219 (1978).
20. A. Mankofsky, T. M. Antonson Jr., and A. T. Drobok, *J. Comput. Phys.*, to be submitted for publication.

Using Adiabaticity for the Limit of Very Low Resistive Simulations*

Michael E. Kress, Ph.D+
Albert A. Blank, Ph.D++

City University of New York
College of Staten Island
130 Stuyvesant Place
Staten Island, New York 10301

*Also at
Science Applications International Corporation
1710 Goodridge Drive
McLean, Virginia 22102

**Also at
Courant Institute of Mathematical Science
New York University
251 Mercer Street
New York, New York 10012
and
Carnegie-Mellon University

ABSTRACT

In a complex plasma topology, most simulation codes for the simplest dissipative model, the Grad-Hogan equations for resistive evolution, have steeply increasing run times as the resistivity tends to zero. Nondissipative (adiabatic) equilibrium simulation codes for the Grad-Shafranov equation are known to be very fast. Seeking to exploit the speed of adiabatic codes and to use it to replace adissipative simulation at very low resistivities, we establish connections between the two limits. The principle link is established by introducing a resistive analog to the adiabatic profile controlling the current density, $\mu(\psi, \tau)$ which is taken as a given datum for the solution of the Grad-Shafranov equation in the low resistivity limit. In effect, this profile defines the thermodynamic state of the adiabatic plasma. The other given adiabatic data, boundary conditions and topological conditions, derive directly from resistive data. We use an oscillating 2D doublet configuration as a test case with a complex x topology. We use the resistive simulation code to study the behavior of the μ profile at low resistivities. As the resistivity decreases, we observe the formation of a boundary layer or sheath around the

separatrix. Outside the region perturbed by the excursion of the separatrix the resistive profile is essentially stationary. This permits precise definition of the sheath region. As the resistivity decreases, dissipative effects are concentrated within the sheath and the sheath thickness decreases rapidly. From this, and our knowledge of other profiles, we infer that the appropriate adiabatic limit is singular, in particular, that the separatrix can become a discontinuity surface that supports jumps in the magnetofluid variables. We conclude that the low resistivity limit may be adiabatic only in a generalized sense. Specifically, the limit is nondissipative in the regions bounded by the separatrix but that the separatrix surface has the possibility of being dissipative by supporting sheet currents and jumps in the MHD variables.

*Work done sponsored in part by Department of Energy Contract No. DE-FG02-86ER53223.

A Method for Computing Fields Near the Origin
of a Cylindrical Coordinate System
in Time Domain Finite-Difference
Electromagnetic Simulations

M. I. Pasak,* D. B. Seidel, M. L. Kiefer, R. S. Coats,
T. D. Poulton, and J. P. Quintenz
Sandia National Laboratories
Albuquerque, New Mexico

One problem with electromagnetic time domain finite-difference simulations in cylindrical coordinates is the rapidly decreasing characteristic dimension of the cells as r approaches zero. In order to satisfy the Courant stability condition a small time step is needed to insure stability, which is undesirable because it increases the cost of the simulation. In our presentation we will describe a method which uses a rectangular grid and an annular cylindrical grid which overlap to perform electromagnetic simulations of cylindrical geometries. The two grids are connected by interpolating the field at the grid points of one grid using field values from the second grid.

The proposed method has been implemented in the QUICKSILVER¹ family of computer codes (QUICKSILVER, the MERCURY preprocessor, and various postprocessing software). QUICKSILVER is a multitasked, finite difference, three dimensional, fully relativistic, electromagnetic, particle in cell code developed at Sandia National Laboratories. In order to conserve central memory, QUICKSILVER avoids discretizing large conductor volumes by employing logically connected simulation regions or "blocks". The computational overhead required to perform the block connections in the field solving routines accounts for a small percentage of the total processing time. The block connection algorithm used in QUICKSILVER exchanges the field quantities between physical cells and block buffer cells in connecting blocks. Block buffer cells are non physical cells that are used to transfer field values between connected blocks.

In the proposed method, buffer cells play an analogous role in the algorithm for connecting the cylindrical and rectangular blocks. In the cylindrical (r, ϕ, z) block, the buffer cells

* The permanent address of M. I. Pasak is the Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ 85721

consist of the first row of block buffer cells which form an annular shell on the inside surface of the annular cylindrical block. In the rectangular (x, y, z) block, the buffer cells are the cells that extend past the buffer cells in the cylindrical block and form a rectangular shell. After each field-solve step the field values are passed from the physical cells in one grid to the buffer cells in the other grid. The presence of two different coordinate systems complicates the exchange of field values because the field points in the two blocks do not necessarily coincide and the vector components depend on the coordinate system. We overcome these difficulties by linearly interpolating the field from the two field locations in the physical cells that bracket the actual field location in the buffer cell. Currently the bracketing grid points are found at every time step. Eventually a more efficient method such as finding and storing the bracketing grid points in a typical plane will be implemented. In addition, the field components must be converted when passed between grids with different coordinate system which requires the evaluation of transcendental functions at each time step. Again this process can be speeded up by precomputing the values prior to beginning the time step loop. In our presentation, we will try to address the pros and cons of using this method in terms of the additional computational overhead associated with the interpolation and conversion of the field components.

We have limited our initial investigation to electromagnetic simulations but the extension of the method to particle simulations is relatively straightforward. Algorithms for handling particles that cross block boundaries already exist in QUICKSILVER. As was the case with the block connection routines for electromagnetic simulations, some of the particle handling routines will need to be modified for particle simulations.

In order to verify the proposed method, as well as determine its accuracy, we are currently in the process of developing and running a series of benchmark simulations. Preliminary results have been encouraging. The first test case is a perfectly conducting cylindrical cavity which we excite with the azimuthally symmetric $TM_{0,1}$ mode. Because the mode is independent of θ and only excites (E_r, B_ϕ, E_z) , we can also use Ampere's law to modify the

difference equations for the cells at the origin of a purely cylindrical grid to obtain numerical results for a cylindrical grid. Results from simulations using a cylindrical grid and a hybrid cylindrical-rectangular grid are in good agreement. To excite the cavity we have used two methods. The first method sets the tangential electric field on the center r -plane of the cavity and simulates half of the cavity. An absorbing boundary condition is also implemented at the excitation plane. In the second method we load the field at every point at $t = 0$ and observe the field as time progresses. Again we only simulate half the cavity but this time by requiring mirror symmetry at the center r -plane of the cavity. We are now looking at ways of exciting the cavity which will better measure the dispersion characteristics of the hybrid cylindrical-rectangular grid. The new method will excite the cavity using a small aperture or the equivalent dipole sources¹. In addition to azimuthally symmetric cavity modes we will also consider asymmetric modes. A second test case will compare theoretical and computed values for a line source and a point source located inside a cylindrical waveguide.

References

- 1 D. B. Seidel, M. L. Kiefer, R. S. Coats, A. L. Siegel, and J. P. Quintenz, Proc. 12th Conf. on Numerical Simulation of Plasmas, San Francisco, CA, paper PT-24 (1987).
- 2 R. E. Collin, *Field Theory of Guided Waves-2nd ed.*, (IEEE Press, Piscataway, NJ, 1991), p. 499

Dispersion and Stability of Two Electromagnetic Solvers for Triangular Meshes

P. W. Rambo
University of California, Lawrence Livermore National Laboratory
Livermore, CA 94550

Time domain electromagnetic solvers on nonorthogonal grids is an area of current research interest. This interest stems from the need to simulate complex geometries in the fields of both plasma physics and electromagnetic scattering. Some of the approaches being investigated include curvilinear coordinates, finite volumes, and finite elements on meshes of quadrilaterals or triangles, both structured and unstructured [1-3]. Most workers have concentrated on algorithm development, with test problems serving as benchmarks. Very little analytic work has been reported concerning stability and accuracy of these methods, even on regular meshes. This paper summarizes an investigation of a limited set of schemes for solving Maxwell's equations for TE polarization on a two dimensional mesh of triangles. Two particular realizations are the weighted residual finite element algorithm of Ambrosiano et al. [1], and differencing based on Delaunay-Voronoi meshes [2]. A dispersion relation is derived for the case of a regular skewed mesh, which includes equilaterals as a special case. As a complement, numerical dispersion tests have been performed for grids generated by randomly perturbing a regular grid.

The field quantities are the z -component of the magnetic field B_z belonging to the j -th triangle and the electric field in the x - y plane at each triangle side s , $\vec{E}_s \cdot \hat{e}_s$, where \hat{e}_s is the unit vector tangent to the side. A limited set of algorithms may be defined by assuming leap frog time advance and restricting the spatial discretization to,

$$\frac{B_j^{n+1/2} - B_j^{n-1/2}}{\Delta t} = -\frac{c}{A_j} \sum_{s \in j} d_s E_s^n (\hat{e}_s \times \hat{n}_j) \cdot \hat{z} \quad (1)$$

$$\frac{E_s^{n+1} - E_s^n}{\Delta t} = -\frac{(B_j^{n+1/2} - B_j^{n-1/2})}{c} \frac{h_j}{A_j} \quad (2)$$

Here, A_j is the area of triangle j , d_s is the length of side s , and h_s is the "distance" between B_j and $B_{j'}$, the magnetic fields in adjacent triangles. The summation in Eq. (1) is over sides belonging to triangle j , with the correct sign determined by the vector product $\hat{e}_s \times \hat{n}_j$ with \hat{n}_j defined as the unit vector normal to side s pointing outward from triangle j . The notation $B_{j'}$ ($B_{j''}$) in Eq. (2) refers to triangles on the positive (negative) $\hat{e}_s \times \hat{e}_j$ side. This class of solvers exactly conserves the divergence constraints in vacuum (within roundoff), is fast, and easy to implement. Furthermore, assuming time dependence $\exp(-i\omega t)$, and eliminating E_s in favor of B_j results in a matrix eigenvalue system, which may be symmetrized, ensuring that all frequencies are purely real or purely imaginary. Ambrosiano and co-workers [1] have proposed a scheme of this form based on weighted residual finite elements, in which case the geometric length "connecting" magnetic field values is defined by $h_s = 2(A_j + A_{j'})/3d_j$. An alternative [2] prescription is based on the Delaunay triangulation

and its dual Voronoi mesh with magnetic fields placed at the Voronoi points (the Voronoi points are defined by the intersection of the side perpendicular bisectors). In this instance the length h_s is defined by the length of the appropriate Voronoi polygon side. Other possibilities exist, such as defining h_s as the distance between triangle centroids, or mixing the two prescriptions above.

The accuracy and stability of both the weighted-residual (WR) and Delaunay-Voronoi (DV) schemes are analyzed for a regular skewed triangular mesh, as shown in Fig. 1(a). The coordinate system is defined by the unit vectors $\hat{e}_1 = \hat{x}$ and $\hat{e}_2 = \cos\theta\hat{x} + \sin\theta\hat{y}$, where $\cos\theta = \hat{e}_1 \cdot \hat{e}_2$ defines the skewing angle θ . The plane may be periodically filled with parallelogram unit cells, labeled by indices (l, m) , located at $\vec{r}_{lm} = m\Delta\eta\hat{e}_1 + l\Delta\xi\hat{e}_2$, as shown in Fig. 1(b). This periodicity allows standard dispersion analysis to be applied, assuming solutions of the form $\exp(-i\omega t + ik_1\Delta\xi + ik_2\Delta\eta)$, where $k_1 = k \cdot \hat{e}_1$ and $k_2 = k \cdot \hat{e}_2$ are wave vector components in the skewed coordinate system. Then the dispersion relation is found to be of the general form,

$$\sin^2\left(\frac{\omega\Delta t}{2}\right) = \frac{\Omega^2}{2} \left[1 \pm \sqrt{1 - \Psi} \right] \quad (3)$$

Here, Ω is a constant defined by the mesh parameters and Ψ contains the wave vector dependence. For small wave number ($k_2\Delta\xi \ll 1$ and $k_1\Delta\eta \ll 1$), $\Psi \approx k^2$ and two approximate solutions are readily found. The lower sign gives the expected physical mode, while the upper sign gives a high frequency branch.

$$\omega = \omega_- = \frac{\Omega}{\Delta t} \sqrt{\Psi} \approx k \quad \omega = \omega_+ = \frac{2}{\Delta t} \sin^{-1} \Omega \quad (4)$$

This is completely analogous to the dispersion relation for phonons in a non-Bravais crystal lattice; the low (high) frequency mode has $B_{lm} \sim B_{lm}$ ($B_{lm} \sim B_{lm}$) similar to the phase of ion displacement for an acoustic (optical) phonon. The consequences of this high frequency branch for particle simulations are not known at this time.

For the case of equilaterals, $\Delta\xi = \Delta\eta = \Delta$ and $\theta = \pi/3$, both the WR and DV schemes are identical. The dispersion relation for this special case is given by,

$$\sin^2\left(\frac{\omega\Delta t}{2}\right) = 3v^2 \left[1 \pm \sqrt{1 - \frac{4}{9} \left\{ \sin^2\left(\frac{k_2\Delta}{2}\right) + \sin^2\left(\frac{k_1\Delta}{2}\right) + \sin^2\left(\frac{k_2\Delta - k_1\Delta}{2}\right) \right\}} \right] \quad (5)$$

Expanding this result for small wavenumber, shows second order accuracy, $\omega = ck$. The stability constraint is easily calculated, $v\omega\Delta t/\Delta < 1$, $\Delta/6 = 0.4082$, being most stringent for the upper sign in Eq. (5) at $k=0$. Some solutions of $\omega(k)$ are shown in Fig. 2, for various propagation directions ($v=0.3$).

Dispersion relations for both WR and DV methods have been derived for arbitrary $\Delta\xi/\Delta\eta$ and θ . In the long wavelength limit the DV method still gives second order accuracy, with $\omega = ck$; in contrast, the WR dispersion relation gives a phase speed which is dependent on the propagation direction. As an example, for the case of right square triangles, $\Delta\xi = \Delta\eta = \Delta$ and $\theta = \pi/2$, $\omega/ck = (3/2)^{1/2} [1 - \sin(2\phi)/3]^{1/2}$, where ϕ is the

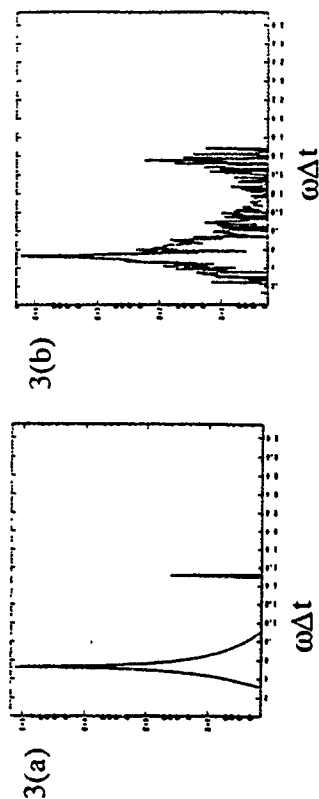
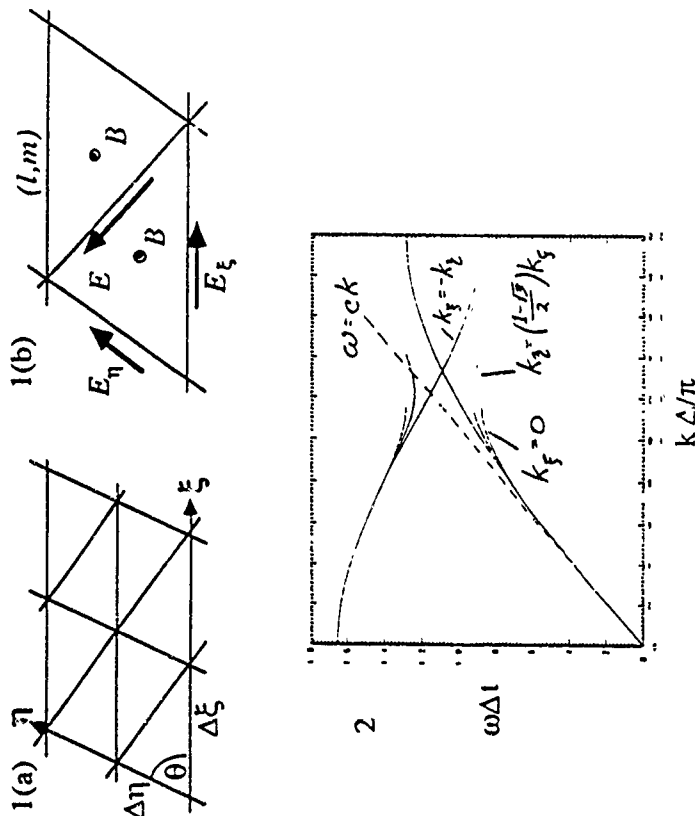
propagation angle measured relative to \hat{k} . It should be pointed out, however, that the DV scheme falls for this mesh, since the Voronoi points are located on the hypotenuses ($h=0$) and stability requires $\Delta\omega=0$. The better phase properties of the DV scheme on this regular mesh stem from the fact that the DV scheme is perfectly centered since the triangle sides perpendicularly bisect the Voronoi sides and vice versa. The WR prescription, however, is more robust.

An opposite extreme of the regular grid analyzed above is the case of a mesh generated by randomly perturbing an initially regular mesh. Figure 3(a) shows the Fourier spectrum of B from a simulation on a 64×64 periodic grid of regular equilaterals with $v=0.3$. The fields were initialized with $k_x=0$, $k_y=2\pi/(4\Delta)$. The observed peaks in the spectrum agree with the analytically determined frequencies $\omega/\Delta=0.531$ and $\omega/\Delta=1.513$. Figure 3(b) shows results for the WR method for the same problem but with the vertices randomly perturbed with uniform distribution $-0.105(6x, 6y) \leq 0.10$. Very similar results are observed for the DV method. If the maximum perturbation is increased to $\Delta r/\Delta=0.2$, the WR scheme still gives similar results while the DV scheme is no longer stable. For longer wavelengths, the spectral peaks are sharper with less noise as might be expected due to more effective averaging of the perturbations. The absence of observable spectral shift relative to the unperturbed solution indicates effective second order accuracy.

Two recently proposed methods for solving Maxwell's equations on triangular grids have been investigated. Dispersion relations for both methods have been derived for the case of a regular skewed grid, which includes equilateral triangles as a special case. Numerical dispersion tests have been carried out for the case of random perturbations to a regular equilateral grid. Although somewhat artificial, the skewed mesh analysis is locally representative of grids obtained in practice, particularly for boundary-fitted logically structured meshes. For the grid analyzed here, the WR prescription suffers from poor dispersion properties when the mesh is skewed from equilateral. The real promise for geometric flexibility, however, lies in unstructured meshes for which the perturbed equilateral grid investigated here might be more representative. On this perturbed grid, both methods appear to be effectively second order with the WR prescription being more robust than the DV prescription. A hybrid approach seems practical, using the DV definitions in areas where the mesh is smoothly varying and the WR definition in regions where the DV prescription would result in an overly restrictive stability condition.

The author wishes to acknowledge valuable conversations with J. J. Ambrosiano, S. T. Brandon, S. L. Ray, and A. B. Langdon. This work was performed under the auspices of the U. S. Department of Energy by Lawrence Livermore National Laboratory under contract number W-7405-Eng-48.

- [1] J. J. Ambrosiano, S. T. Brandon and R. Lohner, "A New Weighted Residual Finite Element Method for Computational Electromagnetics in the Time Domain", 7-th Annual Review of Progress in Applied Computational Electromagnetics, March 19-21, 1991.
- [2] Niel K. Madsen and Richard W. Ziolkowski, "Numerical Solution of Maxwell's Equations in the Time Domain Using Irregular Nonorthogonal Grids", Wave Motion 10, 583 (1988).
- [3] Michael E. Jones, "Electromagnetic PIC Codes with Body-fitted Coordinates", *Proc. Twelfth Conf. Num. Sim. Sim. Plasmas*, San Francisco, Sept. 20-23 1987.



APPROXIMATION OF THE VLASOV-MAXWELL SYSTEM EXPANDED IN FOURIER SERIES, BY A FINITE VOLUME METHOD COUPLED TO A PARTICLE SIMULATION.

F. PAYERDON* - F. DELAURENS** - F. HERMELINE***

* ADIRA BP N°48 St Médard en Jallas, FRANCE

** CEA, CESTA, BP N°2 33114 Le Barp, FRANCE

*** CEA, CELV 94195 Villejuive St Georges cedex, FRANCE

As part of the study of Free Electron Laser (FEL), we handle the problem of relativistic charged particle beam propagation in 3D structures. We inject a relativistic electron beam slightly off the axis of a cylindrically symmetric waveguide, which is imbedded in an external electromagnetic field. We want to compute self-consistently the field radiated by the beam in the waveguide: thus we have to solve a 3D Vlasov-Maxwell system.

Because this would be very expensive, we make the assumption that the field components and the charge and current densities can be expanded in Fourier series with respect to the azimuthal coordinate θ .

Since the beam is slightly off the axis, very few modes are needed to describe the physical phenomenon. Thus the problem comes to the resolution of a small number of 2D (r, z) Maxwell systems, coupled to a 3D Vlasov equation. We solve each of these 2D systems by a finite volume technique on an unstructured Delaunay-Voronoi mesh. This method is numerically coupled to a particle simulation of the Vlasov equation.

The backbone of the finite volume technique we present is the use of two polygonal meshes on the 2D computational domain: a Delaunay mesh and a Voronoi mesh. They show the following property: a vector tangent to a side of a Delaunay Polygon (DP) is normal to the side of a neighbouring Voronoi Polygon (VP) and vice-versa.

Thus using either the DP or the VP when integrating, the finite volume formulation we get is a differential system with the same degrees of freedom for the fields in the different Maxwell equations.

We shall now detail this process. For each Fourier mode indexed by m , the 2D Maxwell system writes

$$\left\{ \begin{array}{l} \frac{\partial \vec{B}_{m\theta}}{\partial t} + \text{curl } \vec{E}_m = 0 \quad (1-a) \\ \frac{1}{c} \frac{\partial \vec{E}_m}{\partial t} - \text{curl } \vec{B}_{m\theta} + \text{jm} \left(\frac{\vec{B}_{m\theta}}{r} - \frac{\vec{D}_{m\theta}}{r} \right) = -\mu_0 \vec{J}_m \quad (1-b) \\ \text{Div } \vec{E}_m + \frac{\text{jm}}{r} \vec{E}_{m\theta} = \frac{\rho_m}{\epsilon_0} \quad (1-c) \end{array} \right. \quad \text{with } \vec{E}_m = (\vec{E}_{m\theta}, \vec{E}_{mz})$$

$$(11) \left\{ \begin{array}{l} \frac{1}{c} \frac{\partial \vec{E}_{m\theta}}{\partial t} - \text{curl } \vec{B}_m = -\mu_0 \vec{J}_m \quad (11-a) \\ \frac{\partial \vec{B}_m}{\partial t} + \frac{1}{r} \text{curl } \vec{E}_{m\theta} - \text{jm} \left(\frac{\vec{E}_{m\theta}}{r} - \frac{\vec{D}_{m\theta}}{r} \right) = 0 \quad (11-b) \\ \text{Div } \vec{B}_m + \frac{\text{jm}}{r} \vec{B}_{m\theta} = 0 \quad (11-c) \end{array} \right. \quad \text{with } \vec{B}_m = (\vec{B}_{m\theta}, \vec{B}_{mz})$$

assuming for the charge and current densities and all the field components a Fourier decomposition of the form :

$$A(r, \theta, z) = \sum_{m=-\infty}^{+\infty} A_m(r, z) e^{jm\theta}$$

To obtain the finite volume formulation of system (11), we achieve the following steps :

- integrate equation (1-a) over a DP in r and z .
- integrate equation (1-b) over a side of VP with weight r , after multiplying scalarly by the unit vector normal to this side.
- integrate equation (1-c) over a VP.
- proceed the same way for system (II), swapping the roles of DPs and VPs.
- and obtain :

$$\left\{ \begin{array}{l} \frac{\partial}{\partial t} \int_{\text{DP}} \vec{B}_m \cdot d\vec{r} - \sum_{\text{sides of DP}} \vec{E}_m \cdot \vec{\tau} d\sigma = 0 \quad \text{for all DPs} \\ - \frac{\partial}{\partial t} \int_{\text{side of VP}} \vec{E}_m \cdot \vec{\tau} d\sigma + c^2 \int_{\text{side of VP}} \vec{\nabla}(r \vec{B}_{m\theta}) \cdot \vec{\nu} d\sigma - \text{jm} c^2 \int_{\text{side of VP}} \vec{B}_m \cdot \vec{\nu} d\sigma \\ = \frac{1}{\epsilon_0} \int_{\text{side of VP}} \text{jm} \cdot \vec{\tau} d\sigma \quad \text{for all VP sides} \\ - \sum_{\text{sides}} \int_{\text{side of VP}} \vec{E}_m \cdot \vec{\tau} d\sigma + \text{jm} \int_{\text{VP}} \vec{E}_{m\theta} dr dz = \frac{1}{\epsilon_0} \int_{\text{VP}} \rho dr dz \quad \text{for all VPs} \end{array} \right. \quad \left\{ \begin{array}{l} \frac{\partial}{\partial t} \int_{\text{VP}} \vec{E}_{m\theta} dr dz + c^2 \sum_{\text{sides}} \int_{\text{side of VP}} \vec{B}_m \cdot \vec{\nu} d\sigma = -\frac{1}{\epsilon_0} \int_{\text{VP}} \vec{J}_\theta dr dz \quad \text{for all VPs} \\ \frac{\partial}{\partial t} \int_{\text{side of DP}} \vec{B}_m \cdot \vec{\nu} d\sigma - \int_{\text{side of DP}} \vec{\nabla}(r \vec{E}_{m\theta}) \cdot \vec{\tau} d\sigma + \text{jm} \int_{\text{side of DP}} \vec{E}_m \cdot \vec{\tau} d\sigma = 0 \\ \sum_{\text{sides}} \int_{\text{side of DP}} \vec{B}_m \cdot \vec{\nu} d\sigma + \text{jm} \int_{\text{DP}} \vec{B}_{m\theta} dr dz = 0 \quad \text{for all DPs} \end{array} \right.$$

where $\vec{\nu}$ and $\vec{\tau}$ are respectively the outgoing normal and the tangent to the side of a DP

Then we make the following assumptions :

- E_{m0} is constant over every DP;
- B_{m0} is constant over every VP;

- $E_m \cdot \tau$ and $B_m \cdot v$ are constant along each DP side and its neighbouring VP side

Thus the unknowns of the problem are the quantities E_{m0} , $E_m \cdot \tau$, B_{m0} and $B_m \cdot v$.
The time differential system involving these degrees of freedom is then discretized with a centered leap-frog scheme

As for the electron beam transport, we suppose it is collisionless, and we describe it with a kinetic model : the Vlasov equation gives a statistical description of the electrons, involving their distribution function $f(x, v, t)$ it writes :

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + \frac{F}{m_e} \cdot \nabla_v f = 0$$

where m_e is the electron mass, and F is the Lorentz force resulting from the electromagnetic field. We use a particle method to solve this equation numerically : the distribution function is approximated by a sum of Delta functions in the phase space, the particles :

$$f(x, v, t) = f_0(x, v, t) = \sum_{i=1}^N w_i \delta(x - x_i(t)) \delta(v - v_i(t))$$

This method accurately computes the convection terms because the particles move along the characteristic of the convective field, while the initial data is approximated in the same way as f .

The charge and current densities are respectively the zeroth and first order moments of the distribution function f_0 . On one hand Maxwell equations involve the Fourier modes of these densities on the Eulerian Delaunay mesh ; on the other hand, Vlasov equation provides the values of the distribution function at the time dependent particles localities, that is on a Lagrangian mesh.

Going from the particle description of the distribution function to the mesh quantities involved in Maxwell equations is called the assignment process : we regularize the zeroth and first order moments of the distribution function, using piecewise linear functions defined on the Delaunay mesh

Conversely, the beam dynamics depends on the values of the electric and magnetic fields on the Lagrangian mesh, whereas Maxwell equations provide the values of their modes on the Eulerian Delaunay mesh. The transition from these latter quantities to the former ones is called the interpolation process : in a first step, we interpolate the field modes at the particles location in the (x, z) plane, and in a second step we reconstruct the 3D fields by summing their Fourier modes.

Finally, let us notice that the zeroth order moment of Vlasov equation gives a relation between the charge and current densities, the continuity equation :

$$\frac{\partial \rho}{\partial t} + \text{Div } j = 0$$

Therefore, solving Vlasov-Maxwell system in a consistent way requires compatibility between this equation and the discretized Vlasov-Maxwell system. For that purpose, we are led to correct Poisson equation at each time step : namely E_m is fitted in order to match a charge density satisfying the continuity equation

REFERENCES

- [BIR] C.K. BIRDSALL, A.B. LANGDON : "PLASMA PHYSICS VIA COMPUTER SIMULATION", Mac Graw-Hill company (1985).
- [BRA] BRACKBILL, COIJEN : " MULTIPLE TIMES SCALES " Academic press (1985)
- [CAI] Z. CAI : " On the finite volume element method ", Numerical Math 58, pages 713-735 (1991)
- [CHIO] S. CHOUHURY and R.A. NICOLAIDES : " Discretisation of incompressible vorticity-velocity equations on triangular meshes ", International Journal for Numerical Methods in Fluids, vol 11, pages 822-833 (1990)
- [DEG] P. DEGOND, F. HERMELINE, P.A. RAVIART, J. SEGRE : " A coupled particle finite-element method for the Vlasov-Maxwell equations. " manuscript 1990
- [HER] F. HERMELINE : " Two coupled particle-finite element methods using Delaunay-Voronoi meshes for the approximation of the Vlasov-Poisson and Vlasov-Maxwell equations - Rapport CEA/CELV/DMA/MCN
- [HER2] F. HERMELINE : " Deux schémas d'approximation des équations de Vlasov-Maxwell bidimensionnelles sur les mailages de Delaunay-Voronoi " Rapport CEA-N-2591, CELV Mars 1989.
- [HOC] R.W. HOCKNEY, J.W. EASTWOOD : "COMPUTER SIMULATION USING PARTICLES" Adam Hilger, Bristol and Philadelphia (1988)
- [MAD] N.K. MADSEN, R.W. ZIOLKOWSKI : " Numerical solution of Maxwell's equations in the time domain using irregular nonorthogonal grids ", Wave Motion 10, pages 583-596 (1988)
- [NIC] R.A. NICOLAIDES : " Direct discretisation of plasmas div-curl problems ", ICASE report 89-76 (oct 1989)

A relativistic Eulerian Vlasov code for the numerical Simulation of the laser-plasma beat-wave current drive

A. Ghizzo,^{*} P. Bertrand^{*}, M. Shoucri,
T. Johnston,[†] M. Feix,[‡] E. Fijaalkow,[§]
S. Karttunen,[¶] T. Patschkangas,^{||} R. Salomaa[§]

Centre Canadien de Fusion Magnétique,
Varennes, P.Q., Canada
J3X 1S1

We present an Eulerian Vlasov Code for the numerical study of the laser-plasma beat-wave current drive. In this model, the plasma is described by "fluid" equations in the direction normal to the external magnetic field, while keeping its kinetic features in the direction of the external magnetic field. Similar methods have been previously used without magnetic field [1]. The electron distribution function $f(x, p_x, t)$ describing particle motion along x -direction satisfies the 1D relativistic Vlasov equation

$$\frac{\partial f}{\partial t} + \frac{p_x}{m\gamma} \frac{\partial f}{\partial x} + e \left(E_x + (\vec{u}_1 \times \vec{B}_1)_x \right) \frac{\partial f}{\partial p_x} = 0 \quad (1)$$

where γ denotes the Lorentz factor: $\gamma = (1 + p_x^2/m^2 c^2)^{1/2}$.

The term $(\vec{u}_1 \times \vec{B}_1)_x = u_y B_z - u_z B_y$ represents the x -component of the Lorentz force. The transverse velocity \vec{u}_1 is obtained through the introduction of the canonical momentum variable defined as follows

$$\vec{P}_1 = m \vec{u}_1 + e \vec{A}_1 \quad (2)$$

which satisfies the following macroscopic (fluid) evolution equation

$$\left(\frac{\partial}{\partial t} + v_x \frac{\partial}{\partial x} \right) \vec{P}_1 = \vec{P}_1 \times \vec{\omega}_c - e \vec{A}_1 \times \vec{\omega}_c \quad (3)$$

The potential vector \vec{A} is in the perpendicular (transverse) direction i.e. $\vec{A} = \vec{A}_1$ and it follows that the electric field can be written in the form

^{*}Physique Théorique, Université de Nancy, France
[†]INRS Énergie, Univ. du Québec, Canada
[‡]PMMS/CNRS-Université d'Orléans, France
[§]Helsinki University, Finland

$$\vec{E} = \vec{E}_{||} + \vec{E}_\perp \text{ where } E_{||} = E_x = \frac{-\partial \phi}{\partial x} \text{ and } \vec{E}_\perp = \frac{-\partial \vec{A}_1}{\partial t} \quad (4)$$

and

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{e}{\epsilon_0} (n_e - n_e(z, t)) = 0 \quad (5)$$

n_e is the homogeneous ion density background and $n_e(z, t)$ the electron density given by:

$$n_e(z, t) = \int f(x, p_x, t) dp_x \quad (6)$$

We consider the propagation of a circularly polarized light in a plasma. The TEM fields E_y , B_z and E_z , B_y obey Maxwell's equations, which can be written in the following form (with $E^\pm = E_y \pm i B_z$, $F^\pm = E_z \pm i B_y$):

$$\frac{\partial E^\pm}{\partial t} \pm c \frac{\partial E^\pm}{\partial x} = -J_y / \epsilon_0 \quad (7)$$

$$\frac{\partial F^\pm}{\partial t} \mp c \frac{\partial F^\pm}{\partial x} = -J_z / \epsilon_0 \quad (8)$$

These equations are integrated along their vacuum characteristics. $x \pm ct = \text{constant}$ [1]. J_y and J_z denote the y and z component of the transverse current density defined by

$$\vec{J}_\perp(x, t) = en_e(x, t) \vec{u}_1(x, t) \quad (9)$$

The numerical algorithm used to solve these equations consists of a method of fractional steps, with a scheme space centered and time centered, similar to what has been previously presented [1].

The initial values of TEM fields are given by:

$$\begin{aligned} E_y(x, t=0) &= E_0 \cos k_0 x \\ E_z(x, t=0) &= \epsilon E_0 \sin k_0 x \end{aligned} \quad (10)$$

$\epsilon = +1 (-1)$ corresponds respectively to a right (left) circularly polarized wave. The magnetic field can be written in the form:

$$\begin{aligned} B_y(x, t=0) &= -\epsilon \frac{k_0 E_0}{\omega_0} \sin k_0 x \\ B_z(x, t=0) &= \frac{k_0 E_0}{\omega_0} \cos k_0 x \end{aligned} \quad (11)$$

The corresponding mean transverse velocity is then defined by:

$$\begin{aligned} u_y(z, t=0) &= \frac{-e}{m} \frac{E_0}{\omega_p - \epsilon \omega_c} \sin k_z z \\ u_z(z, t=0) &= \frac{e}{m} \frac{E_0}{\omega_p - \epsilon \omega_c} \cos k_z z \end{aligned} \quad (12)$$

$\omega_c = e B_0/m$ denotes the electron cyclotron frequency. The corresponding values of the canonical momentum components are then:

$$\begin{cases} P_y(z, t=0) = -\frac{e E_0}{\omega_p(\omega_p - \omega_c)} \sin k_z z \\ P_z(z, t=0) = \frac{e E_0}{\omega_p(\omega_p - \omega_c)} \cos k_z z \end{cases} \quad (13)$$

and for the potential vector components

$$\begin{cases} A_y(z, t=0) = \frac{E_0}{\omega_p} \sin k_z z \\ A_z(z, t=0) = -\frac{E_0}{\omega_p} \cos k_z z \end{cases} \quad (14)$$

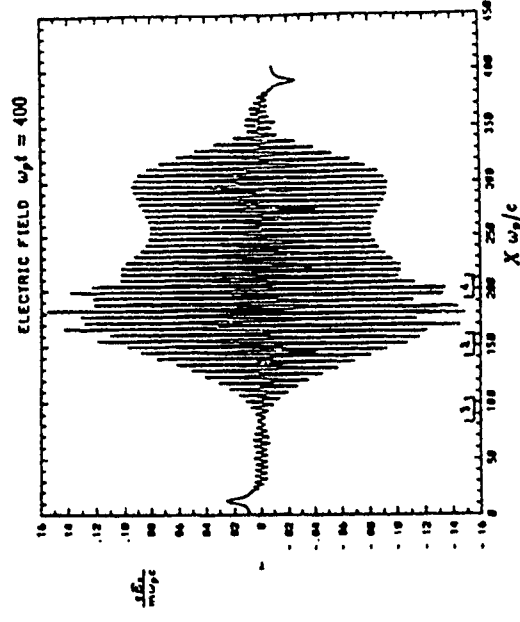
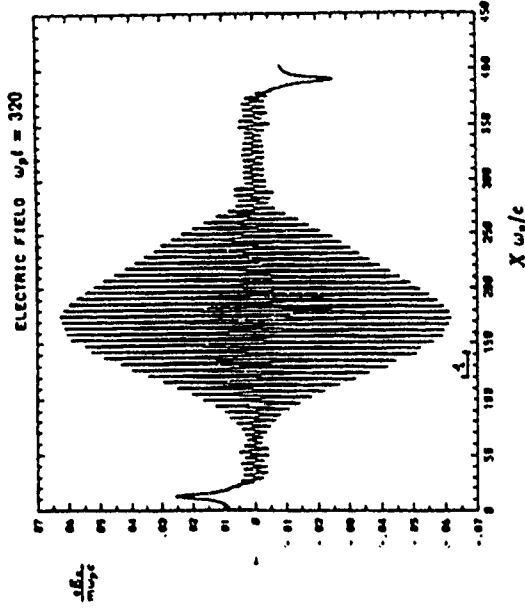
The initial electron distribution function is given by

$$f(x, p_x, t=0) = \frac{1}{\sqrt{2\pi} v_{te}} e^{-\frac{1}{2} \left(\frac{p_x}{v_{te}} - \alpha \cos k_z x \right)^2}$$

where $\alpha \cos k_z x$ is an initial perturbation with $\alpha = 0.2$ and k_z is the plasma wavenumber. The simulation is effected in a bounded code, of total length $L = 276.33 \lambda_D$. A right-hand circularly polarized wave is penetrating at the left of this system, and propagating to the right. The intensity of the wave is $E_0 = 0.12074$, and $V_{0,ex} = E_0/\omega_p = 0.02$. The total number of points in x -space is $N = 2432$, $\Delta x = L/N$ and $c \Delta t = \Delta x$. The simulation up to $t = 520 \omega_p^{-1}$ takes a CPU time of 3288 sec with 2198 time steps. With 512 points in velocity space, this is equivalent to 2432×512 "particles", and correspond to $1.2 \mu s$ /time step per grid point. The pump wave, with a wavenumber of $k_z c/\omega_p = 4.3006$ and frequency $\omega_p = 4.4721$ is exciting a plasma wave through the forward Raman scattering process $k_s = k_z + k_r$, $\omega_s = \omega_c + \omega_r$, with $\omega_r = (\omega_p^2 + 3 k_r^2 V_A^2)^{1/2} = 1.079 \omega_p$ for 20 keV plasma with $V_A = 0.1978$. This corresponds to a phase velocity $V_{\phi} = \omega_r/k_r = 0.9122$. The scattered wave is such that $\omega_s = 3.393 \omega_p$ and $k_s c/\omega_s = 3.1178$. The external magnetic field is such that $\omega_c = 1.5 \omega_p$. The parameters are taken from [2] and corresponds to ITER like parameters. The figures show the excited plasma wave at $t = 320 \omega_p^{-1}$ and $t = 400 \omega_p^{-1}$, as the electromagnetic wave penetrates the plasma from left to right.

References

- [1] P. Bertrand, A. Ghizzo, M. Shoucri, T. Johnston, E. Fijalkow, M. Feix, J. Comp. Phys. 90, 431 (1990).
- [2] S. Karttunen, T. Pättikangas, R. Salomaa, to be published, Nuclear Fusion.



NUMERICAL SOLUTION OF THE DRIFT-KINETIC VLASOV EQUATION

M Shoucri, A Ghizzo*, P. Bertrand*, M Feix† E Fijaalkow†

Centre Canadien de Fusion Magnétique, Varennes, J3X 1S1, P.Q., Canada

The problem of $\vec{E} \times \vec{B}$ drift at a tokamak edge is of great importance in tokamak physics. It is now generally believed that the anomalous particle and energy transport in tokamaks is caused by turbulent fluctuations generated by instabilities due to the $\vec{E} \times \vec{B}$ drift in the plasma edge. Also, $\vec{E} \times \vec{B}$ rotation associated with a radial electric field in tokamak edge have been associated with improved confinement performance of the tokamaks known as R-mode [1].

We present a numerical code and preliminary results in studying the solutions of an important equation pertinent to this problem, the drift-kinetic Vlasov equation or guiding center equation. The advantage of the Vlasov codes over particle in-cell codes for such a problem is obvious. Given the low noise level of these codes, they provide a powerful tool to study low frequency oscillations. This equation couples the motion along a magnetic field with the $\vec{E} \times \vec{B}$ motion normal to the magnetic field (Eqs. 1, 2). We solve these equations for the case of a slab model, periodic in the x-direction, finite in the y-direction, and allows the total magnetic field \vec{B} (in the $x-z$ plane) to have an angle θ with the z direction. In this system of coordinates, the electron distribution function $f(x, y, v_{||e})$ for the electrons obey the following equation:

$$\frac{\partial f_e}{\partial t} + \left(v_{||e} \cos \theta + c E_y \frac{\sin \theta}{B} \right) \frac{\partial f_e}{\partial z} - E_x \frac{c \sin \theta}{B} \frac{\partial f_e}{\partial y} - \frac{e E_x \cos \theta}{m_e} \frac{\partial f_e}{\partial v_{||e}} = 0 \quad (1)$$

and the ions distribution function $f(x, y, v_{||i})$ obey the following equation:

$$\frac{\partial f_i}{\partial t} + \left(v_{||i} \cos \theta + E_y \frac{c \sin \theta}{B} \right) \frac{\partial f_i}{\partial z} - E_x \frac{c \sin \theta}{B} \frac{\partial f_i}{\partial y} + \frac{e E_x \cos \theta}{m_i} \frac{\partial f_i}{\partial v_{||i}} = 0 \quad (2)$$

We solve the Poisson equation:

$$\Delta \varphi = \frac{e}{\epsilon_0} (n_e(x, y, t) - n_i(x, y, t)) \quad (3)$$

and

*Physique Théorique Univ. de Nancy, Nancy, France
†IPN/MS/CNRS-Université d'Orléans, Orléans, France

$$E_x = -\frac{\partial \varphi}{\partial x}, E_y = -\frac{\partial \varphi}{\partial y} \quad (4)$$

n_e and n_i denote respectively the electron and ion density given by.

$$n_e(x, y, t) = \int f_{e,s}(x, y, v_{||e,s}) dv_{||e,s} \quad (5)$$

The numerical code developed applies a method of fractional steps associated with a cubic spline interpolation technique [2]

The Numerical Scheme

The fractional steps method applied to Eq (1) (and to Eq (2)) involved five integration steps.

A1 Between t_n and $t_{n+1/2}$ we shift the electron distribution function in z -space for a time $\Delta t/2$ and we obtain:

$$f_e^*(x, y, v_{||e}, t_{n+1/2}) = f_e \left(x - \left(v_{||e} \cos \theta + c E_y^{n+1/4} \frac{\sin \theta}{B} \right) \frac{\Delta t}{2}, v_{||e}, t_n \right) \quad (6)$$

A2 We compute $E_x^{n+1/4} = E_x(x, y, t_{n+1/4})$ and we shift the distribution function f_e along the x -axis for a time $\Delta t/2$

$$f_e^{**}(x, y, v_{||e}, t_{n+1/2}) = f_e^* \left(x - \left(v_{||e} - E_x^{n+1/4} \frac{\sin \theta}{B} \right) \frac{\Delta t}{2}, v_{||e}, t_{n+1/2} \right) \quad (7)$$

A3 We compute the electric field components at time $t_{n+1/2}$ from Poisson's equation and we shift the distribution function in $v_{||e}$ we a full time step Δt :

$$f_e^{***}(x, y, v_{||e}, t_{n+1/2}) = f_e^{**} \left(x, y, \left(v_{||e} - \frac{e}{m_e} E_x^{n+1/2} \cos \theta \right) \Delta t, t_{n+1/2} \right) \quad (8)$$

A4 We shift again the distribution function in y -space for half time step

$$f_e^{****}(x, y, v_{||e}, t_{n+1}) = f_e^{***} \left(x, y - E_y^{n+3/4} \frac{\sin \theta}{B} \frac{\Delta t}{2}, v_{||e}, t_{n+1/2} \right) \quad (9)$$

A5 We shift f_e in the z -direction for a time $\Delta t/2$

$$f_e(x, y, v_{||e}, t_{n+1}) = f_e^{****} \left(x - \left(v_{||e} \cos \theta + E_y^{n+3/4} \frac{\sin \theta}{B} \right) \Delta t, y, v_{||e}, t_{n+1/2} \right) \quad (10)$$

The components of the electric field \vec{E} are calculated as follows.

$$\epsilon_0 \frac{\partial \vec{E}}{\partial t} = -\vec{J} = -\vec{J}_{||} - \vec{J}_{\perp} \quad (11)$$

$$\vec{J}_{||} = -e \int_{-\infty}^{\infty} f_e(x, y, v_{||e}, t) v_{||e} dv_{||e} + e \int_{-\infty}^{\infty} f_i(x, y, v_{||i}, t) v_{||i} dv_{||i} \quad (12)$$

$$\vec{J}_{\perp} = -e c \frac{\vec{E} \times \vec{B}}{B^2} (n_e(x, y, t) - n_i(x, y, t)) \quad (13)$$

$$E_z^{n+1/4} = E_z^{n-1/4} + \frac{\Delta t}{2} \frac{e}{\epsilon_0 B} (n_e(x, y, t_n) - n_i(x, y, t_n)) - e \cos \theta J_{||}^n \Delta t / 2 \quad (14)$$

$$E_y^{n+1/4} = E_y^{n-1/4} - \frac{\Delta t}{2} \frac{e}{\epsilon_0 B} (n_e(x, y, t_n) - n_i(x, y, t_n)) \quad (15)$$

At half time-steps, the electric field is calculated from Poisson's equation. Since the x is periodic, a Fourier transform in the variable x in Poisson equation is performed:

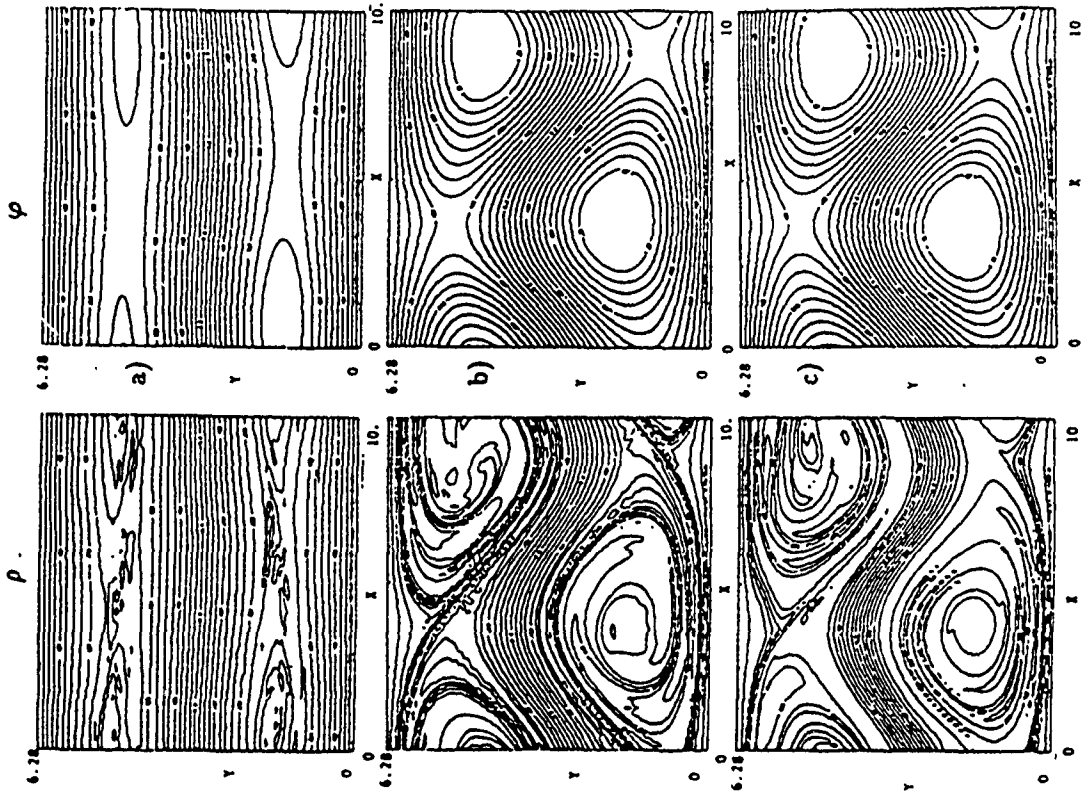
$$-k^2 \varphi_k(y, t_{n+1/2}) + \frac{\partial^2}{\partial y^2} \varphi_k(y, t_{n+1/2}) = -\rho_k(y, t_{n+1/2}) \quad (16)$$

$$\rho(x, y, t_{n+1/2}) = -\frac{e}{\epsilon_0} (n_e(x, y, t_{n+1/2}) - n_i(x, y, t_{n+1/2})) \quad (17)$$

As a test for the code, we present in the figure the results obtained in the special case of $\theta = \frac{\pi}{4}$, with a velocity profile $U_0 = -\bar{e} \cdot \nabla \varphi = \cos y \bar{e}_y$. In this case, the instability in due to the velocity shear and the linearized vorticity equations (Rayleigh stability equation) give a growth rate of 0.122, [3] in very good agreement with the numerical value of 0.125 obtained with the present code. The present code allows to study the nonlinear evolution of this instability until saturation. The contour plots in the figure show the vorticity ρ and the stream function φ at: a) $\omega_p t = 100$; b) $\omega_p t = 140$; c) $\omega_p t = 200$ until saturation. Results for $\theta \neq \pi/2$ will also be presented for instabilities due to velocity shear.

References

- [1] R. Weynants, in Plasma Physics and Controlled Fusion Proceeding of the IAEA Conference (Washington, Octobre 1980) to be published.
- [2] A. Ghizzo, P. Bertrand, M. Shoucri, T. Johnston, M. Feix, E. Fijaalkov, J. Comp. Phys. 90, 43 (1990); M. Shoucri and R. Gagné, J. Comp. Phys. 27, 315 (1978)
- [3] M. Shoucri and G. Knorr, Plasma Phys. 18, 204 (1976); M. Shoucri Int. J. Num. Methods Eng. 17, 1525 (1981).



SLOW-TIME-SCALE SIMULATION OF RF DEVICES

H. P. Freund
Science Applications International Corp., McLean, VA 22102

The application of slow-time-scale finite element analyses of coherent RF devices has received a great deal of attention over the past several decades, and the current generation of simulation codes has proven to be both computationally efficient and accurate predictors of experiment for a wide range of devices. The general approach to be described involves the representation of the electromagnetic field in terms of a superposition of modes supported by the general structure of the device. The analysis is both three-dimensional and inherently multi-mode. In the cases described herein, we are concerned with the interaction in various waveguide geometries.

For a coherent narrow bandwidth device, a single-frequency approximation can be made which permits Maxwell's equations to be averaged over a wave period. As a consequence, the fast time scale behavior can be eliminated from the analysis, resulting in a significant reduction in the computational requirements. The present paper is concerned with the description of three types of devices: cyclotron masers,^{1,2} free-electron lasers,^{3,4} and Cerenkov masers.⁵ This approach permits the derivation of slow-time-scale equations describing the evolution of the amplitude and phase of each mode in terms of a generalized source current found by a particle average. The equations are not device-specific in the sense that the interaction is controlled by the particle motion, and the slow-time-scale equations can be applied to any of these devices. The detailed physics of the interaction mechanism is determined by the form of the fields used in the Lorentz force equations. Thus, for example, the equations for the amplitudes and phases of an ensemble of modes in a cylindrical waveguide can be applied to the study of either a free-electron laser or a cyclotron maser depending upon the form of the magnetostatic fields and the initialization of the electron beam.

We first consider an ensemble of TE modes in a loss-free cylindrical waveguide, which can be represented in the form

$$\delta A(x, t) = \sum_{l=0}^{\infty} \sum_{n=1}^{\infty} \delta A_{ln}(z) \left[\frac{J_l(\kappa_{ln} r)}{\kappa_{ln} r} \right] \hat{e}_r \sin \alpha_{ln} + J_l'(\kappa_{ln} r) \hat{e}_\theta \cos \alpha_{ln} \quad (1)$$

where for frequency ω and wavenumber $k_{ln}(z)$

$$\alpha_{ln} = \int_0^z dz' k_{ln}(z') + l\theta - \omega x \quad (2)$$

denotes the phase, J_l and J_l' denote the regular Bessel function of the first kind and its derivative, and κ_{ln} describes the cutoff wavenumber of each mode. Here $\kappa_{ln} = x_{ln}/R_1$, where $J_l'(x_{ln}) = 0$ and R_1 is the waveguide radius. The TM modes can be treated in a similar fashion. It is implicitly assumed that the mode amplitudes $\delta A_{ln}(z)$ and wavenumbers $k_{ln}(z)$ vary slowly over a wavelength.

The dynamical equations for the amplitude and phase are obtained by substitution of this representation for the field into Maxwell's equations. The specific procedure involves an average over the wave period and subsequent orthogonalization of the equations in r and θ . This projects out the equations for each specific mode, which become

$$\left[\frac{d^2}{dz^2} + \left(\frac{\omega^2}{c^2} - k_{ln}^2 - \kappa_{ln}^2 \right) \right] \delta A_{ln} = \frac{\omega^2}{c^2} H_{ln} \left\langle \frac{v_1 T_{ln}^{(1)} + v_2 W_{ln}^{(1)}}{|v_3|} \right\rangle \quad (3)$$

$$2k_{ln}^{1/2} \frac{d}{dz} \left(k_{ln}^{1/2} \delta A_{ln} \right) = \frac{\omega^2}{c^2} H_{ln} \left\langle \frac{v_1 W_{ln}^{(-)} - v_2 T_{ln}^{(-)}}{|v_3|} \right\rangle \quad (4)$$

Here $T_{ln}^{(s)} = F_{ln}^{(s)} \sin \psi_{ln} + G_{ln}^{(s)} \cos \psi_{ln}$, $W_{ln}^{(s)} = F_{ln}^{(s)} \cos \psi_{ln} - G_{ln}^{(s)} \sin \psi_{ln}$, (v_1, v_2, v_3) are the components of the velocity in a rotating frame in which $v_r = v_1 \cos k_{\omega} z - v_2 \sin k_{\omega} z$, $v_y = v_1 \sin k_{\omega} z + v_2 \cos k_{\omega} z$, and $v_z = v_3$, $H_{ln} = x_{ln}/|x_{ln}|^2 - l^2/|r|^2(x_{ln})$,

$$\psi_{ln} = \psi_0 + \int^t dz' [k_{ln} + lk_{\omega} - \frac{d\theta}{dz'}] \quad (5)$$

is the phase, $F_{ln}^{(s)} = J_{l-1}(\kappa_{ln} r) \cos[(l-1)\chi] \pm J_{l+1}(\kappa_{ln} r) \cos[(l+1)\chi]$, $G_{ln}^{(s)} = J_{l-1}(\kappa_{ln} r) \sin[(l-1)\chi] \pm J_{l+1}(\kappa_{ln} r) \sin[(l+1)\chi]$, and $\chi = \theta - k_{\omega} z$. Finally, the particle average appearing in Eqs. (3) and (4) is defined for a monoenergetic beam with a pitch angle spread as follows

$$\langle (\cdots) \rangle = \frac{A}{4\pi\lambda_g} \int_0^{2\pi} d\theta_0 \int_0^{p_0} dp_{\perp 0} \beta_{\perp 0} \exp[-(p_{\perp 0} - p_0)^2/\Delta p_{\perp}^2] \times \iint_{\lambda_g} dx_0 dy_0 \sigma_{\perp}(x_0, y_0) \int_0^{2\pi} d\psi_0 \sigma(\psi_0) (\cdots) \quad (6)$$

where

$$A = \left[\pi \int_0^{p_0} dp_{\perp 0} \exp[-(p_{\perp 0} - p_0)^2/\Delta p_{\perp}^2] \right]^{-1} \quad (7)$$

is a normalization constant, $\beta_{\perp 0} = v_{\perp 0}/c$, $\psi_0 = \tan^{-1}(p_{\perp 0}/p_{\parallel 0})$, and $p_{\perp 0}$ defines the initial axial momentum of the electron. It is important to recognize that this average implicitly includes the effect of the overlap of the electron beam with the transverse mode structure of the radiation field [often included in one-dimensional analyses in an *ad hoc* manner by the inclusion of a filling-factor] in a self-consistent manner.

Observe that the preceding equations were derived specifically to treat a free-electron laser with a helical wiggler field. In this case, the electrons describe helical trajectories with a wavenumber k_{ω} . However, these equations do not include any of the interaction physics and can be readily transformed to x, y, z coordinates and used to describe either free-electron lasers or cyclotron masers.

As mentioned, the above formulation is multi-mode in the sense that an arbitrary number of modes may be treated using the dynamical equations. Coupling between the modes is included through the effect of the mode ensemble on the particle trajectories. Hence, in order to complete the formulation, the electron orbit equations in the presence of the static and fluctuation fields must be specified. Since we deal with an amplifier model we choose to integrate in z and write the Lorentz force equations in the form

$$v_i \frac{d}{dz} p = -e \delta E - \frac{e}{c} v \times [B_{ext}(x) + \delta B] \quad (8)$$

where δE and δB represent the electric and magnetic fields associated with the specified vector potential δA , and $B_{ext}(x)$ is the externally applied magnetostatic field.

The nonlinear differential equations are solved numerically for a configuration in which a wave of frequency ω is injected into the system at $z = 0$ in concert with the electron beam. The solution to this initial value problem is accomplished by a fourth-order Runge-Kutta-Gill technique. The averages are performed by means of an N th order Gaussian quadrature technique in each of the variables $(r_0, \theta_0, \psi_0, \phi_0)$. In the absence of energy spread, this reduces to (r_0, θ_0, ψ_0) and the choice of $N_r = N_\theta = N_\psi = 10$ (i.e., 1000 particles) is typically made. The inclusion of energy spread requires additional electrons. In this case, a reduction in the number of electrons in the radial positions to $N_r = 4$ does not result in a severe loss of accuracy, and we choose 1, 6th and 4th order Gaussian quadrature to resolve the p_0 and ϕ_0 distributions [for a total number of 9600 electrons].

We now consider the specific case of a free-electron laser with a helical wiggler field.³ The initial state of the electron beam is chosen to model the injection of a monoenergetic, uniform, axisymmetric electron beam with a flat-top density profile (i.e., $\sigma_L = \sigma_0 = 1$). Electron positions are chosen by means of the Gaussian algorithm within the ranges $-\pi \leq \psi_0 \leq \pi$, $0 \leq \theta_0 \leq 2\pi$, and $R_{min} \leq r_0 \leq R_{max}$. In the absence of an energy spread, the momenta are chosen so that $p_{L0} = 0$ and $p_{r0} = m_e c (\gamma_0^2 - 1)^{1/2}$, where γ_0 is the relativistic factor corresponding to the total beam energy.

The nonlinear formulation described herein has been compared with an intense beam free-electron laser experiment conducted at the M.I.T. with good results.⁶ This experiment employed a helical wiggler field with a period of 3.14 cm, an amplitude of up to almost 2 kG, and an entry taper region six wiggler periods in length. No axial magnetic field was imposed. The electron beam was produced by a PI Pulsar 110 A accelerator in concert with a field-emission electron gun to produce a beam with a pulse length of 30 nsec., a maximum energy of up to 2.5 MeV, and a maximum current of up to 7 kA. The beam was then focused and apertured to produce a high-quality beam with a current of up to approximately 1 kA, which propagated through a drift tube with an inner diameter of 0.8 cm. The output of the free-electron laser was found to range in frequency from approximately 200-500 GHz at power levels of the order of 18 MW. Estimates of the axial energy spread gave approximately $\Delta\gamma/\gamma_0 = 0.72 - 0.40\%$.

Measurements of the growth rate were made using a kicker magnet which dumped the electron beam to the wall at various points. Two examples were for wiggler amplitudes of 1510 G and 1275 G respectively for a beam energy of 2.0 MeV and current 780 A. The gains for these cases were 70 ± 6 dB/m and 56 ± 3 dB/m respectively. The variation in the gain with axial energy spread for these cases was studied using the numerical simulation. For the case of $B_w = 1275$ G, the calculated gain is found to be approximately 62 dB/m for $\Delta\gamma/\gamma_0 = 0$, and decreases to 56 dB/m for $\Delta\gamma/\gamma_0 = 0.25\%$. Similarly, for $B_w = 1510$ G, the gains calculated for $\Delta\gamma/\gamma_0 = 0$ and $\Delta\gamma/\gamma_0 = 0.25\%$ are 70 dB/m and 64 dB/m respectively. These results are consistent with the interpretation of an energy spread of the order of 0.25%. This is in excellent agreement with the estimates of beam quality obtained from the electron gun. Hence, the value of $\Delta\gamma/\gamma_0 = 0.25\%$ is used henceforth.

The measured output spectrum for the case of a 2.3 MeV, 930 A electron beam and a wiggler field amplitude of 1275 G is shown in Fig. 1, together with the results of the nonlinear simulation. Total measured output power is of the order of 18 MW for this example. In this case, the simulation included the TE₁₁ and TM₁₁ modes. Since the nonlinear formulation is restricted to single frequency propagation, the numerical procedure used to generate the theoretical curve was to consider each frequency separately and to plot the combined output spectrum normalized to the peak power. As can be seen in the figure,

the spectral agreement is excellent, and the measured and computed spectral peak differed by less than 2%.

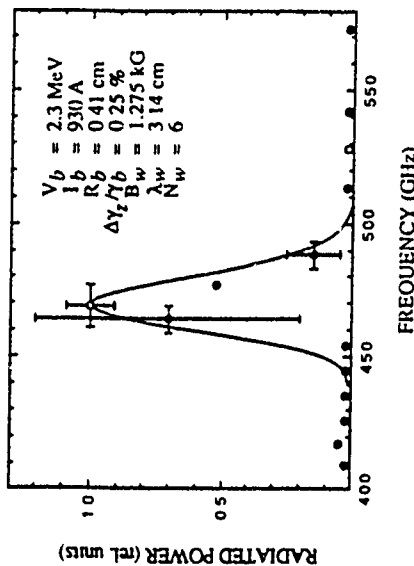


Fig. 1 Comparison of the output spectrum from experiment and theory. The dots represent the measurements.

Similarly good agreement with experiment is found for analogous codes written to treat planar wigglers,⁴ cyclotron masers,⁵ and Cerenkov Masers.³ The run times required for these codes been studied for the case of 1000 electrons propagating through 1250 steps on a Cray X-MP. This is adequate to simulate the free-electron laser experiment at LLNL.⁷ The run time was found to scale to be of the order of $\tau = 0.15N + 0.22$, where τ is the run time in minutes and N is the number of modes in the simulation. As a result, it is possible to simulate a wide range of practical RF devices with a minimal allocation of computer resources. Finally, it is concluded that there is no necessity to employ large-scale PIC simulation techniques for a wide range of RF device applications.

The author would like to acknowledge the collaboration of Dr. A.K. Ganguly in the development of these techniques. This work was supported by the Office of Naval Research and the Office of Naval Technology.

REFERENCES

- ¹A.K. Ganguly and S.Y. Ahn, *Int. J. Electron.* **53**, 641 (1982).
- ²H.P. Freund and C. Chen, *Int. J. Electron.* (submitted for publication).
- ³A.K. Ganguly and H.P. Freund, *Phys. Rev. A* **32**, 2275 (1985).
- ⁴H.P. Freund, *Phys. Rev. A* **37**, 3371 (1988).
- ⁵H.P. Freund, *Phys. Rev. Lett.* **65**, 2993 (1990).
- ⁶D.A. Kirkpatrick, G. Bekefi, A.C. DiRienzo, H.P. Freund, and A.K. Ganguly, *Phys. Fluids B* **1**, 1511 (1989).
- ⁷T.J. Orzechowski, B. Anderson, J.C. Clark, W.M. Fawley, A.C. Paul, D. Prosnitz, E.T. Schaefermann, S. Yarema, D.B. Hopkins, A.M. Sessler, and J.S. Wurtele, *Phys. Rev. Lett.* **57**, 2172 (1986).

1 and 2 Species Fluid Simulations of the Solar Wind AMPTE Release

J. B. Harold and A. B. Hassam

(Laboratory for Plasma Research, University of Maryland, College Park, MD 20742-3511).

The AMPTE solar wind release of December 1984 exhibited an unexpected shift of the Barium cloud transverse to the solar wind flow, followed by a downwind acceleration of the cloud (see figure 1). By employing simplified fluid models¹ we have reproduced the essential features of this event using numerical simulations (see figure 2). The fluid models are essentially MHD models generalized to allow time scales on the order of or shorter than the ion gyroperiod. We show that this model is derivable from first principles provided typical flow speeds are supersonic, as is the case for the solar wind AMPTE release.

The direct approach to a numerical simulation of this problem is a hybrid one, treating electrons as a magnetized fluid and ions as single particles. However, these simulations are numerically intensive and have been performed only for very early times after release. By using a fluid approach, we are able to perform longer simulations and examine a variety of different parameter regions.

We are currently working with two simulation codes. The first treats only a single ion species (one electron fluid, one ion fluid). With it we have obtained qualitative agreement with the gross behavior of the AMPTE experiment². Preliminary runs from a second code which allows for two ion species confirm these results. Each of these simulations is explicit, fourth order in space, and second order in time³.

In each of these codes we find that three parameters govern the magnitude of the transverse shift: the Alfvén Mach number, the relative mass of the cloud, and the ratio p/R , where p is the effective ion Larmor radius and R is the cloud size. The shift itself is the result of the generation of a magnetic field on the $-u \times B$ side of the cloud. This field, the result of the inclusion of the Hall term in the fluid equations, exerts a pressure on the cloud. In response the bulk of the cloud shifts in the $u \times B$ direction, while a smaller part moves at higher velocities in the $-u \times B$ direction, conserving momentum in the system. We are extending this work to examine various parameter regions and the effects of using two fluids of varying masses.

This work has been supported in part by an NRL grant under Contract N00014-89-K-2013.

1. A. B. Hassam and J. D. Huba, *Phys. Fluids* 31, 318 (1988).
2. J. B. Harold and A. B. Hassam, "A Simulation of the December 1984 Solar Wind AMPTE Release", *Geophysical Research Letters*, Vol. 18, No. 2, 135-138 (Feb. 1991).
3. J. Drake, P. Guzdar, A. Dimitz, "3D simulation of grad(T) Driven Turbulence and Transport", *Phys. Fluids*, in press.

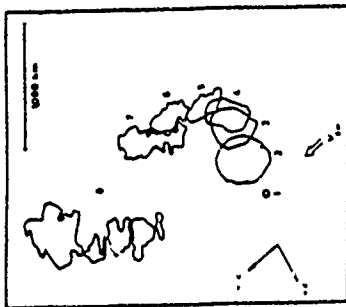


Fig 1) Density contours of the AMPTE release of 27 December 1984. The dotted contour corresponds to the head of the comet (reproduced from Valenzuela et al, 1986)

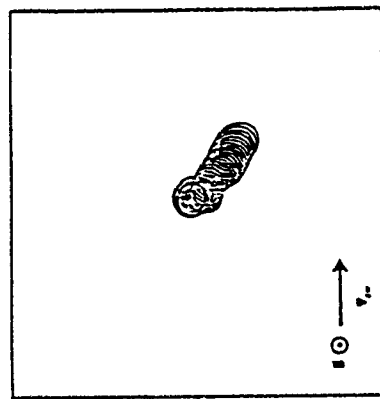


Fig. 2.) A series of density contours (one per time frame) for a simulation run. The magnetic field is out of the page, and the solar wind moves from the left side of the page to right. Note that the cloud first moves in the sub direction (toward the bottom of the page), then turns to move downstream at an angle to the solar wind.

Simulation of Luminosity Enhancement by Colliding Electron and Positron Beams in a Plasma *

C.-K. Ng¹ and S. Rajagopalan² and P. Chen¹

¹Stanford Linear Accelerator Center, Stanford University, Stanford, CA 94309

²Physics Department, University of California, Los Angeles, CA 90024

1. Introduction

To obtain high enough luminosity (event rate) for high energy physics experiments in linear colliders, the colliding e^+ and e^- beams need to be focused to very small sizes at the interaction point. This is crucial for the Stanford Linear Collider (SLC) and for future TeV linear colliders.

Luminosity will be enhanced near the interaction point region by disruption effects, as a consequence of the bending of the particle trajectories in a beam under the influence of the electromagnetic fields provided by the other oncoming beam. To further achieve luminosity enhancement, a plasma lens¹ has been proposed to focus the individual colliding e^+ and e^- beams in a plasma before they reach the interaction point. The basic idea is to make use of the pinching effects by the large transverse wake fields generated by the interaction between a relativistic electron or positron beam with the plasma. The individual beams are then focused before they collide at the interaction point. More recently, it is proposed² that an optimal enhancement can be achieved by placing a thick plasma lens around the interaction point so that beams are both focused and collide inside the plasma.

In this paper, we use the 2-D PIC code CONDOR to simulate beam-beam collisions in a background plasma. To simulate this scheme, we modified CONDOR by inserting a luminosity calculation. In the next section, we show the results for the focusing of e^+ and e^- beams in a plasma. In section 3, we discuss the luminosity enhancement for e^+e^- collisions by focusing and colliding them in a plasma.

2. Plasma lens focusing

The density profile for a cylindrically symmetric bi-Gaussian beam is

$$\rho_L = n_L e^{-r^2/2\sigma_r^2} e^{-z^2/2\sigma_z^2}, \quad (1)$$

where the peak beam density n_L is related to the number of particles N in the bunch as $n_L = N/(2\pi)^{3/2} \sigma_r^2 \sigma_z$. The electric fields of an incoming beam are strong enough to ionize a gas into a plasma³ which will then focus the beam. This is particularly useful since the production of a high density plasma of $n_p \sim 10^{18} \text{ cm}^{-3}$ near the interaction point region

To avoid aberration of beam optics and background for particle detectors, an underdense lens for focusing a beam is investigated. The beam is chosen to be denser than the plasma and typically $n_L \sim 2n_p$. In this regime, the total refraction of the plasma electrons by the e^- beam produces a uniform ion column and the electrons will be focused linearly. For e^+ beam, the plasma electrons behave differently and the focusing is not as linear.

* Work supported by DOE contracts DE-AC03-76SF00515 and DE-AS03-88ER40381.

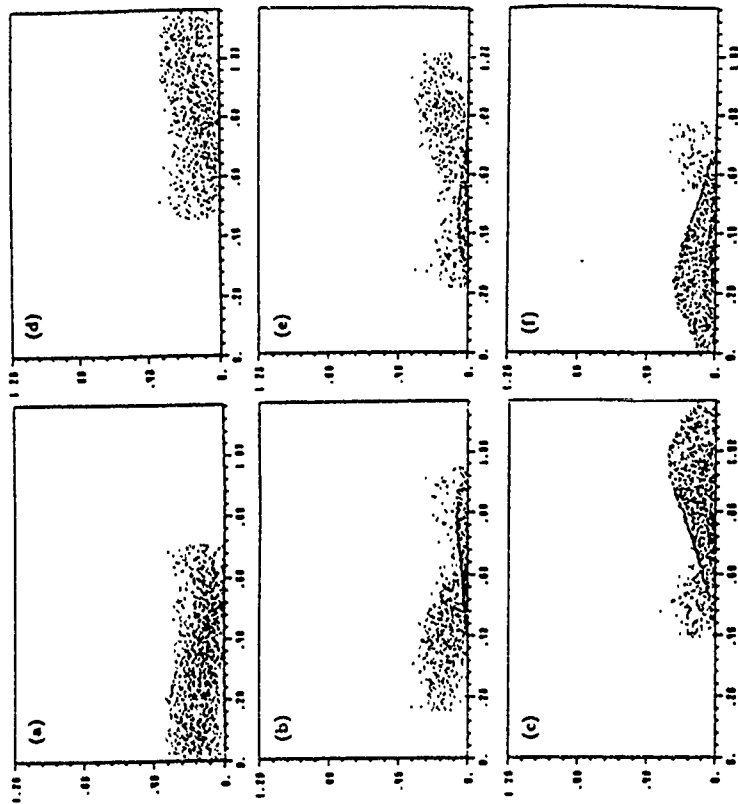


Fig. 1 Focusing and colliding of e^+ and e^- beams in a plasma. (a)-(c) are the distributions of electrons at three different time steps: 2.3 psec, 3.2 psec and 4.0 psec. (d) and (f) are those of positrons for the same corresponding time steps. The horizontal axis z and the vertical axis r are in units of mm and 10 μm respectively.

For the simplicity of simulation, a uniform plasma is formed initially before the beams are injected into the plasma region. In Fig. 1, we show the results from a CONDOR run for the behaviors of the e^+ and e^- beams as they traverse the plasma. The initial conditions are $n_p = 3 \times 10^{18} \text{ cm}^{-3}$, $n_L = 4.8 \times 10^{18} \text{ cm}^{-3}$, $\sigma_r = 0.2 \text{ mm}$ and $\sigma_z = 1.4 \mu\text{m}$ with beam energy $E = 5 \text{ GeV}$ ($\gamma = 1 \times 10^4$). The electrons are injected from the left boundary and the positrons from the right. The plasma fills the region up to $r = 6 \mu\text{m}$ to mimic the real situation where the plasma is created by tunneling ionization and extends up to a few σ_r 's before the beam collective field drops to a threshold value.^{2,3} Figs. 1(a)-(c) and Figs. 1(d)-(f) are the snapshots at three different time steps for the distributions of e^- and

e^+ beams respectively. From Figs. 1(a) and 1(d), it can be seen that the electrons and positrons are focused gradually by the plasma. The beam fronts are not focused well as the transverse wake fields are not strong enough in this region. For a highly relativistic charged particle beam traveling in vacuum, the self-pinching effects are almost balanced by the space charge forces. However, in a plasma, we clearly see that the beam is pinched and focused. In Figs. 1(b) and 1(e), the beams collide at very small beam sizes at less than $1 \mu\text{m}$. It is important that the beams collide at the focal point of the plasma lens for optimal luminosity gain. In Figs. 1(c) and 1(f), the two beams start to diverge again after colliding with each other and this completes the process of e^+e^- collision in the plasma

3. Luminosity enhancement by plasma lens

We have shown that a plasma can pinch a relativistic e^+ or e^- beam. To ensure that the two beams collide at the focal point of the plasma lens, we implemented a luminosity calculation in CONDOR to monitor the optimal condition for the plasma lens. Theoretically, in the underdense regime, the focal length of the plasma lens is given by²

$$s^* = \frac{\pi}{2\sqrt{K}}, \quad (2)$$

where $K = 2\pi r_e n_p / \gamma$ is the focusing strength of the lens determined by the plasma density n_p . Here r_e is the classical electron radius. If both e^+ and e^- beams are focused evenly, then the plasma lens thickness should be $l = 2s^*$. With various nonlinear effects, however, the desirable lens thickness cannot be determined analytically. We therefore rely on simulation to determine the optimal lens parameters. In CONDOR simulation, the focal length is determined by optimizing the longitudinal length of the plasma for maximum gain in luminosity.

The luminosity for e^+e^- collisions is defined as the 4-dimensional phase space integral

$$\mathcal{L} = f \int n_1(x, y, z, t) n_2(x, y, z, t) dx dy dz dt, \quad (3)$$

where $x' = z - ct$, n_1 and n_2 are the densities of electrons and positrons as functions of time respectively, and f is the repetition rate of collisions. When e^+ and e^- beams collide in vacuum, luminosity will be enhanced because of disruption, as a consequence of the bending of particle trajectories by the electromagnetic fields of the oncoming beam. To quantify this luminosity enhancement, we define the enhancement factor

$$H_{D0} = \frac{\mathcal{L}}{\mathcal{L}_0}, \quad (4)$$

where $\mathcal{L}_0 = f N^2 / 4\pi\sigma_z^2$ is the nominal luminosity. H_{D0} is a function of the dimensionless parameter D , which is a measure of the strength of the pinching and is defined as

$$D = \frac{r_e \sigma_z N}{\gamma \sigma_z^2}. \quad (5)$$

When e^+ and e^- beams collide in plasma, the luminosity is further enhanced because of plasma focusing. An enhancement factor H_D can be similarly defined as in Eq. (4) for the case when the beams collide in the plasma. Thus for a specific disruption parameter D , we can determine the length of the plasma lens by optimizing the enhancement factor H_D .

In Fig. 2, we show the simulation results for the variation of the enhancement factor H_{D0} as a function of D . In our simulations, we only vary N in Eq. (5) and therefore D is directly proportional to N . As expected, H_{D0} increases as D increases. At $D = 1.95$, which corresponds to $N = 6.87 \times 10^{10}$, $H_{D0} = 4.12$. For the beams with the same N but first focused and then colliding in a plasma of length 1.2 mm , the overall enhancement factor $H_D = 11.42$. Comparing with $H_{D0} = 4.12$ for beam collision in vacuum, an extra factor ~ 3 is achieved by plasma focusing. With further investigation, by varying the plasma length, the maximal value of H_D can be determined. This will be useful for designing a plasma lens for final focusing for e^+e^- collisions in linear colliders, e.g., SLAC

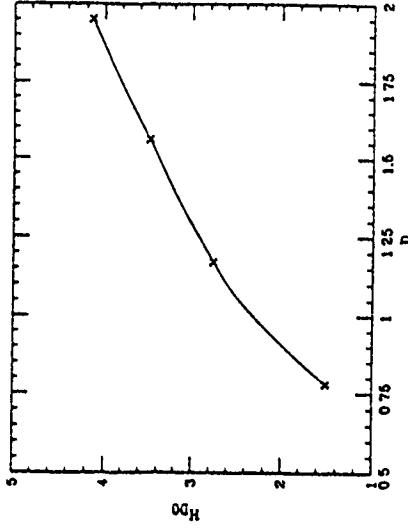


Fig. 2 The enhancement factor H_{D0} as a function of D .

Acknowledgements

We would like to thank K. Eppley and K. Ko of SLAC for their help on CONDOR simulation.

References

- [1] P. Chen, Particle Accelerators 17, 121 (1987).
- [2] P. Chen, SLAC-PUB-5502 (1991).
- [3] S. Rajagopalan, C.-K. Ng and P. Chen, these proceedings

A COMPARATIVE STUDY OF PSEUDOSPECTRAL AND FINITE ELEMENT METHODS ON A VELOCITY SHEAR INSTABILITY

S. Ghosh, R. de Fainctsein, E. Siregar
S. Zalesak and N. L. Goldstein
NASA - Goddard, Greenbelt, MD 20771

There are a variety of numerical methods and algorithms available to the computational physicist for studying compressible magnetohydrodynamic (MHD) phenomena in space plasmas. Issues pertaining to turbulence can be studied with spectral codes [1,2] which conserve a variety of analysis invariants while modeling nonlinear cascades over a wide range of spatial and temporal scales. Unfortunately, spectral methods often require periodic boundary conditions and can run into difficulties with shocks and other discontinuities. Realistic boundary conditions and the stable representation of shocks is feasible with finite element-based methods. However, such methods tend not to conserve global quantities such as total energy to as high a precision as spectral methods, and so their applicability for studying nonlinear dynamics over a wide range of spatial scales can be questioned. In this paper we compare the conservation properties and spectral characteristics of runs from three different algorithms for a model problem, the velocity shear instability. The algorithms we consider include a Fourier pseudospectral code [2] which has been constructed with careful attention paid to the conservation of several analytic invariants, a finite element code [3] with the capability of adaptive grid refinements, and a Fourier-Chebyshev pseudospectral code [4] which can handle mixed boundary conditions often encountered in astrophysical plasmas. We have initialized the Fourier pseudospectral and finite element codes with identical initial data and then consider power spectra, contour plots, and global measurements of various quantities after several nonlinear time intervals. We find that even after several eddy turnover times, the finite element code results are remarkably close to those of the pseudospectral code. Finally in a separate but similar comparison, we have studied the Fourier-Chebyshev code against the Fourier-Fourier code.

We consider an MHD system in which the dimensionless time, length, velocity, density, pressure, and magnetic field are expressed, respectively, as t, L, u, ρ, P , and B . The characteristic length scale, L_0 , is related to the characteristic time scale, t_0 , through $t_0 = L_0/U_0$ where U_0 is a characteristic speed. Expressing the MHD equations in terms of these dimensionless quantities we obtain the following forms for the continuity, momentum, and magnetic induction equations

$$\frac{\partial}{\partial t} \rho = -\nabla \cdot (\rho \mathbf{u}), \quad (1a)$$

$$\frac{\partial}{\partial t} \mathbf{u} = -\mathbf{u} \cdot \nabla \mathbf{u} - \frac{1}{\gamma M_0^2} \nabla P + \frac{1}{M_0^2} (\mathbf{J} \times \mathbf{B}) + \frac{1}{\rho} \nabla^2 \mathbf{u} + \frac{1}{\rho} \left(C_1 + \frac{1}{2} C_2 \right) \nabla (\nabla \cdot \mathbf{u}), \quad (1b)$$

$$\frac{\partial}{\partial t} \mathbf{B} = \nabla \times \mathbf{u} \times \mathbf{B} + \mu_1 \nabla^2 \mathbf{B}, \quad (1c)$$

Here the magnetic field, \mathbf{B} , is related to the current, \mathbf{J} , and the vector potential, \mathbf{a} , through $\mathbf{J} = \nabla \times \mathbf{B}$ and $\nabla \times \mathbf{a} = \mathbf{B}$ respectively. The pressure scale is $P_0 = \rho_0 C_s^2 / \gamma$, where C_s is a characteristic sound speed, and ρ_0 is the dimensional density scale. The two constant quantities appearing in (1b) are $M_0^2 = (U_0/C_s)^2$ and $M_0^2 = (U_0/V_A)^2$ where $V_A^2 = B_0^2 / (4\pi \rho_0)$ is the squared Alfvén speed. The last two terms in (1b) represent a familiar model of viscous dissipation, involving viscosity coefficients C_1 and C_2 .

There are several ways to close Eq. (1). We adopt the energy equation for an ideal gas, $P = \rho T$, in the form

$$\frac{\partial}{\partial t} E = -\nabla \cdot \left[(E + \frac{1}{\gamma} P) \mathbf{u} \right] + \frac{1}{2M_0^2} |\mathbf{B}|^2 \mathbf{u} - \frac{1}{M_0^2} (\mathbf{u} \cdot \mathbf{B}) \mathbf{u} + \frac{\mu_1}{M_0^2} \mathbf{J} \times \mathbf{B} - \frac{\kappa_1}{(\gamma-1)M_0^2} \nabla T - \mathbf{u} \cdot \boldsymbol{\tau} \quad (2)$$

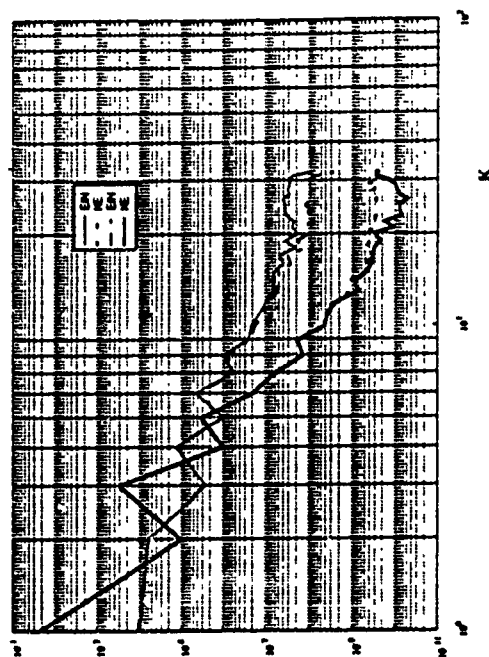
Here, the constant κ_1 is related to the thermometric conductivity, and $\boldsymbol{\tau}$ is the dimensionless temperature stress tensor, $\boldsymbol{\tau}$, is

$$\tau_{ij} = \nu_1 [\partial_i u_j + \partial_j u_i - \frac{2}{3} \partial_k u_k \delta_{ij}] + C_2 \partial_k u_i \partial_k u_j$$

For our first studies as presented in this paper, we consider an unmagnetized, neutral fluid, although we have in mind to later introduce the magnetic field.

We consider the velocity shear instability in a two-dimensional viscous compressible fluid. The simulations are started on a uniform grid of up to 128x128 points, with the fluid at constant density and nearly constant pressure. The initial velocity has a zero y component throughout, and the x component alternates on three horizontal regions between $+U$, $-U$, and $+U$. A small amount of random noise is added to the velocity field to generate the instability, and the viscosity is adjusted to model a state with Reynolds number, $Re = 125$. We have run three sets of simulations by adjusting κ_1 so that the Prandtl number, $Pr = \nu_1 / [(\gamma-1)M_0^2]$, goes from small, moderate, to large.

The qualitative features of a particular class of runs is remarkably similar between the different codes. The power spectra of the kinetic energy (heavy solid line) and the internal energy (dot-dash line) from the pseudospectral code is compared to the power spectra of the kinetic energy (heavy dotted line) and the internal energy (light solid line) from the finite element code in the figure below.



In this case there is no heat conduction and the spectra as shown are after several eddy-turnover times into the simulation. The similarity in the power spectra of the two codes is evident. Only above wavenumbers $k \approx 20$ of this 64x64 resolution run do the spectra begin to deviate substantially.

The temperature at high, intermediate, and low Prandtl numbers ($Pr = 0, 0.00018, 0.012$) also shows a strong similarity between the pseudospectral and finite element codes at equivalent times as illustrated in the set of figures below.



Gas Dynamics, Pseudospectral, $t=30$, $\nu=0$, $.00018, 0.012$



Gas Dynamics, Finite Element, $t=18$, $\nu=0$, $.00018, 0.012$
(Note: Color bars are normalized across each strip independently)

Other detailed comparisons between the Fourier pseudospectral code and the finite element code will be presented. A similar study comparing the Fourier-Fourier and the Fourier-Chebyshev codes will also be presented.

1. D. Gottlieb and S. A. Orszag, *Numerical Analysis of Spectral Methods* (S.I.A.M., Philadelphia, 1977)
2. S. Ghosh, M. Hossain, and W. H. Matthaeus, *Comput. Phys. Commun.* (in preparation)
3. H. Lohner, *Comput. Methods in Appl. Mech. and Engr.* 61:323 (1987)
4. E. Sirkar and M. L. Goldstein, *EOS II 1808* (1990), (Fall AGU meeting, San Francisco, CA).

Particle Simulations with Greatly Reduced Noise

Mike Kotschenreuther
Institute for Fusion Studies
The University of Texas at Austin
Austin, Texas 78712

A new simulation algorithm of general applicability is described which can often greatly reduce noise and particle requirements. The new algorithm is extremely advantageous when the physical fluctuations of the kinetic distribution function δf are often much smaller than the total distribution function f . This is the case for many problems of great interest in plasma confinement devices where there are many or many kinetic micro-instabilities which can be overwhelmed by the noise. The new algorithm can reduce noise in such cases by orders of magnitude, and greatly reduce particle and time requirements. Here, we present simulation results of the new algorithm for the ion gyro-kinetic equation. We note that a similar algorithm was devised independently by Perkins and Tajima previously, however they encountered difficulties and did not pursue it further. Dimitz and Lee have also independently devised a similar algorithm, but did not examine its low noise feature.

Consider a kinetic equation or fluid equation of the form

$$\partial f(\mathbf{x}, t) / \partial t + \partial [U'(\mathbf{x}, t) f] / \partial \mathbf{x}' = 0 \quad (1)$$

where \mathbf{x} is the total particle coordinate, its physical interpretation depends on the physical system. Equation (1) describes a continuum distribution function where the coordinates of each element of f (i.e., particles) change according to

$$d\mathbf{x}/dt = U(\mathbf{x}, t). \quad (2)$$

For example, in the one dimensional Vlasov equation, $X = (x^1, x^2)$, and x^1 is the position, x^2 is the velocity, $U^1(x^1, x^2, t) = x^2$, and $U^2(x^1, x^2, t) = (q/m)E(x^1, t)$, where q/m is the charge to mass ratio and $E(x, t)$ is the electric field. More generally, U depends on fields which are themselves found from f by inverting a differential field equation. For the Vlasov case $dE/dx = q \int f dv$.

In a standard particle in cell algorithm, the continuous distribution function f is replaced by N particles with coordinates \mathbf{x}_i which evolve according to Eq. (2). This type of algorithm will be referred to as a "standard" algorithm.

Noise arises in the standard algorithm which is not present in Eq. (1), since random statistical fluctuations in the number of particles per cell cause field fluctuations.

We now describe the new, low noise particle algorithm. Define $f = f_0 + \delta f$, where $f_0(x, t)$ is an arbitrary function which we call the background distribution. Then

$$\frac{\partial}{\partial t} \delta f + \sum_i \frac{\partial}{\partial x_i} U_i \delta f_i = S(x, t) \equiv -\frac{\partial f_0}{\partial t} - \sum_i \frac{\partial}{\partial x_i} U_i f_0. \quad (3)$$

Note that Eq. (3) is fully equivalent to Eq. (1). We now evaluate $\delta f(x, t)$ at $\mathbf{x} = \mathbf{x}_i(t)$, where $\mathbf{x}_i(t)$ is a solution of Eq. (2),

$$\frac{d}{dt} \delta f(\mathbf{x}_i(t), t) + \left(\sum_i \frac{\partial}{\partial x_i} U_i \right) \delta f(\mathbf{x}_i(t), t) = S(\mathbf{x}_i(t), t). \quad (4)$$

In the low noise algorithm, solution of the ordinary differential equation Eq. (4) provides a direct computation of the perturbed distribution function at the tip of the particle orbit, without counting up the number of particles per cell. Equation (4) is exact for any f_0 , but one tries to choose f_0 make δf as small as possible to minimize noise. The N coordinates $\mathbf{x}_i(t)$ are not relevant as particle positions; they should be thought of as position markers for the value of δf (even though the position markers follow the particle trajectories Eq. (2).)

We consider the case where $\sum \partial U_i / \partial x_i = 0$. This applies when the coordinates \mathbf{x} are canonical coordinates for a single particle Hamiltonian, or for otherwise incompressible fluid flow. Let us start with an initially uniform marker distribution in the region of phase space \mathbf{v}_p . Let us thus say that each marker particle originally "represents" a volume of phase space $v_p = v_p/N$ which surrounds each particle. For incompressible flow, this surrounding volume stays constant in time. Thus, to find the integral of δf over some region of phase space R , (to compute the density of current, for example) we may use the prescription

$$\int_R dx \delta f = \sum_{i \in R} \delta f_i v_p, \quad (5)$$

where the sum includes all markers inside R .

The algorithm outlined above, based on δf , has much less noise than in a standard particle algorithm. Suppose that the statistical average number of particles per cell in a standard algorithm is N_s , with statistical fluctuation in the number of particles per cell of δN . Then the distribution function f has fluctuations of order $\delta N \delta f / N$. Statistical fluctuations of $f = f_0 + \delta f$ in the low noise algorithm are of order $\delta N \delta f / N$; this is a great improvement when f_0 can be chosen to make $\delta f / f_0 \ll 1$. To examine instabilities arising on an equilibrium, the equilibrium is often a good choice for f_0 .

Note that refinements of this scheme have been devised to allow nonuniform particle loading and $\sum \partial v_i / \partial x_i \neq 0$, which will not be described here due to space.

We consider the collisionless electrostatic ion gyrokinetic equation in sheared slab geometry with two spatial dimensions, x and y . In normalized units, the perturbed nonadiabatic distribution function $h(\mathbf{x}, v_\perp, v_\parallel)$ is described by

$$\frac{\partial h}{\partial t} + k y \tilde{\phi} \cdot \nabla h + v_E \nabla h = \frac{\partial \Phi}{\partial t} f_M - v_E \cdot \nabla f_0 \quad (6)$$

where velocity is normalized to $v_i = \sqrt{2T/m}$, the x coordinate is normalized to $\rho_i = v_i m c / q B$ (the ion Larmor radius), time is normalized to the ion diamagnetic frequency (for $k_y = 2\pi/L_y$, where L_y is a chosen periodicity length in the y -direction) and the y coordinate is normalized to k_y^{-1} . Also $\Phi = (q\phi/T)(L_n/\rho_i)$, with L_n the density gradient scale length, and h similarly

normalized by L_n/ρ_i . Define $f_M = e^{-v^2/\pi^{1/2}}$. In these units, the background distribution $f_0 \sim (L_n/\rho_i)f_M$ and $\nabla f_0 = \frac{1}{2} [1 + \eta(v^2 - 3/2)] f_M$. In actual devices $L_n/\rho_i \sim 10^2$; the δf algorithm eliminates the noise from f_0 and makes possible simulations for such realistic parameters.

Additional definitions for Eq. (7) are $k = \frac{1}{2} L_n/L_n$, with L_n the shear length, $v_r = \bar{z} \times \nabla \Phi$, and $\eta_i = \partial \ln T_i / \partial \ln n_i \partial z$. The gyrophase averaging operation is denoted by an overbar, $\bar{\Phi} = \int_0^{2\pi} d\theta \Phi(z + \cos \theta \rho_{\perp}, y + \sin \theta v_{\perp}, k_y \rho_i)$. The field Φ is determined by quasineutrality and using an adiabatic electron response

$$(1 + T_i/T_e)\Phi = 2\pi \int dv_{\perp} v_{\perp} \bar{\Phi}. \quad (7)$$

To avoid the awkwardness of a $\partial \Phi / \partial t$ term in the source of Eq. (4), we rewrite Eq. (7) in terms of $f' = h - \bar{\Phi} f_M$. Equation (2) and Eq. (4) become, for the markers for Eq. (6),

$$\frac{dx_i}{dt} = \bar{z} \times \nabla \bar{\Phi} \quad \frac{dv_{\perp}}{dt} = \frac{dv_{\perp}}{dt} = 0 \quad (8)$$

$$\frac{df_i}{dt} = -\{kxv_{\parallel} + [1 + \eta(v^2 - 3/2)]\} f_M. \quad (9)$$

Note that Eqs. (9) are phase space preserving, so the δf algorithm applies.

We now consider conservation laws. The following energy is conserved by Eq. (7):

$$E_0 = \int dz dv \left\{ \left[h + z f_M [1 + \eta(v^2 - 3/2)] \right]^2 / 2 f_M \right. \\ \left. - f_M \bar{\Phi} \left[1 + \eta(v^2 - 3/2) \right] \right\} - \frac{1}{2} \int dz \left(1 + \frac{T_i}{T_e} \right) \Phi^2. \quad (10)$$

The quantity E_0 , despite its unfamiliar appearance, is in fact a physically meaningful energy, Gibbs free energy. The Vlasov energy of a quasineutral plasma is $E_v = \sum_i \int dz dv \frac{1}{2} m_i v_i^2$, where f'_i is the Vlasov distribution. The entropy $S = \sum_i T_i \int_0^{2\pi} d\theta \int_0^\infty f'_i (f'_i - 1)$ is constant for the collisionless Vlasov equation. Define the free energy $\mathcal{F} = E_v + S$. After expanding \mathcal{F} to second order in the gyrokinetic expansion, one can show that $\mathcal{F} = E_0 + \text{Constant} + \text{H.O.T.}$

For the energy in the δf algorithm, we use the finite particle version of Eq. (10)

$$E_0^c = v_p \sum_i \left[\frac{1}{2} \left(f_i + \left\{ \bar{\Phi}(x_i) + z_i [1 + \eta(v_i^2 - 3/2)] \right\} f_M \right)^2 - f_M z_i \bar{\Phi} [1 + \eta(v_i^2 - 3/2)] \right] \\ - 2\pi L_n \sum_i \frac{\Phi_i^2}{2} \left(1 + \frac{T_i}{T_e} \right) \quad (11)$$

where $\phi = \sum_k \phi_k e^{ikx}$, and L_x is the x length of the simulation volume. Using Eq. (9) - (10) to compute df_0^c/dt , we find

$$E_0^c(t) - E_0^c(0) = \int_0^t dt \sum_i v_p f_M \left(\frac{1}{2} \frac{\partial}{\partial y} \phi^2(x_i) + kx_i^2 \eta_i \frac{\partial \phi(x_i)}{\partial y} \right) [1 + \eta(v_i^2 - 3/2)] \equiv s_E(t). \quad (12)$$

In the limit of an infinite number of evenly distributed particles, $\sum_i v_p Q(x_i, y_i, v_{\perp i}) \rightarrow \int dx dy dv_{\perp} dv_{\parallel} Q(x, y, v_{\perp}, v_{\parallel})$, for any Q . Thus $s_E \rightarrow 0$ in this limit and the continuum energy E_0 is conserved in this limit.

The constancy of $E_0^c(t) - E_0^c(0) - s_E(t)$ requires only that the ordinary differential equations (9)-(10) and Eq. (8) being satisfied. This provides a check on the numerical accuracy of the space and time finite differencing scheme. However, the constancy of $E_0^c(t)$ also depends on the smallness of s_E , i.e., on how closely the simulation approaches the desired infinite particle limit. Thus, examination of both these quantities checks simulation accuracy and also the adequacy of the number of particles used.

A particle code based on Eq. (8)-(12) has been constructed and tested. For linear runs, the δf code agrees with linear fully gyrokinetic eigenvalue codes, typically to within 10 percent.

Comparisons were made with a standard gyrokinetic particle code for the nonlinear saturation of a 2d η , mode in sheared slab. Parameters typical of TFR were used: $\eta_i = 1$ and $L_n/L_s = 0.25$. However, the standard code needed a large equilibrium gradient scale ($\rho_i/L_n = 1/40$, much stronger than experiment) to increase the saturated amplitude above the noise.

The standard particle code was run with 300 K and 3000 K particles. The 300 K simulation was noise dominated; it showed no well-defined exponentiating phase and had a large φ amplitudes. The δf codes used roughly 10 min of Cray CPU time. The δf code gave converged results with 32 K particles ($1/k = 1024$), i.e., energy conservation was good, δf was relatively small and runs with 64 and 128 K particles gave the same saturated amplitude to within 10-15%.

The 3000 K simulation gave a saturated amplitude which agreed well with the δf code, but required more than two orders of magnitude more Cray CPU time. Note that the saturated amplitudes from the particle codes also agreed with results from a fully kinetic spectral code (with f defined on a grid in velocity space) constructed by H.V. Wong and P. Lyter.

All codes need an order of magnitude more time for 3d runs. The δf codes run for acceptable expense (~ 4 Cray CPU hours). The cost of standard gyrokinetic algorithms for such low noise levels would be many hundreds of hours or more.

Friday Morning
September 6, 1991

Invited Oral Session (OF)
8:30am - 1:00pm

Regatta Ballrooms A & B

R. Faehl, presiding

MHD Boundary Layer Calculation Using the Vortex Method

L A Gritzol, K Ganguly†, A.K. De‡, & D. Shent

1. Introduction

Recently, vortex methods have become increasingly attractive for the numerical simulation of incompressible fluid flow at high Reynolds numbers. Primarily based on the technique of Chorin [1], these methods have been applied in various forms to characterize the flow over flat plates [2] and bluff bodies [3], flow with density variations [4], as well as other two and three dimensional flow problems. In this work, we apply the vortex method to simulate the two dimensional boundary layer flow over a flat plate subjected to a transverse magnetic field. A Reynolds Number of 1.0×10^4 was chosen for the analysis. A similarity solution of this problem is given by Rossow [5] in the form of a truncated series. We show that implementation of the vortex method yields results which agree well with the work of Rossow, and provide additional information for conditions adversely affected by truncation of the series solution.

2. MHD Vortex Method

We begin with the momentum equation in the x direction subject to the Prandtl boundary layer assumptions as given by

$$\frac{D\omega}{Dt} = \frac{1}{Re} \frac{\partial^2 \omega}{\partial y^2} + X \quad (1)$$

where, $\omega = \nabla \times \mathbf{U} = -\frac{\partial u}{\partial y}$ and $\frac{D}{Dt}$ is the total (or material) derivative. Also, for an incompressible fluid we have continuity in the form

$$\nabla \cdot \mathbf{U} = 0 \quad (2)$$

The body force term X in Eqn. 1 is the curl of the Lorentz force imposed by the fluid elements crossing the magnetic field lines. For low magnetic Reynold numbers and constant properties, the body force term for a fluid of density ρ and conductivity σ can be written in the form

$$X = m \frac{\partial u}{\partial y} \quad (3)$$

where, $m = \frac{\sigma B_0^2}{\rho}$. For this problem, we have the following boundary conditions (1) $u = v = 0$ at $y = 0$, (2) $u = U_\infty$ for $x \leq 0$, (3) $\frac{\partial u}{\partial y} = v = 0$ as $y \rightarrow \infty$, and (4) $\frac{\partial \omega}{\partial y} = -m$ as $y \rightarrow \infty$

† Texas Tech University, Department of Mechanical Engineering.

‡ Texas Tech University, Department of Mathematics.

The vortex method is applied through the use of vortex sheets. As is customary in vortex methods, we solve the Euler equation in Lagrangian coordinates to determine the discretized evolution of vortex sheet strength. For our case, the Euler equation given by

$$\frac{D\omega}{Dt} = -m\omega \quad (4)$$

is written along the particle trajectories for $\mathbf{x}(\alpha, 0) = \alpha$ as

$$\frac{D\omega}{Dt}(\mathbf{x}(\alpha, t), t) = -m\omega(\mathbf{x}(\alpha, t), t) \quad (5)$$

and has the solution

$$\omega(t + \Delta t) = \omega(t)e^{-m\Delta t}. \quad (6)$$

With the evolution of vorticity and the freestream velocity known, the velocity (u, v) can be determined using the method of Chorin [2]. In our case, the freestream velocity varies according to boundary condition (4) as given by

$$u_\infty(x) = -mx + U_\infty. \quad (7)$$

The vortex method then follows as given in the paper by Chorin [2] and the program by Cheer [6], where the viscous part of Eqn. 1 is addressed through the use of the random walk in the y direction only.

For a constant free stream velocity ($m = 0$), Chorin [2] employs the momentum defect formula to obtain estimates of the coefficient of skin friction. As our problem is subject to the variable freestream velocity given by Eqn. 7, we focus our attention on obtaining estimates of the displacement thickness and the velocity profile.

3. Displacement Thickness

To establish the validity of the simulation, the displacement thickness calculated by the vortex method was compared to that given by the similarity solution of Rossow [5]. Comparisons were performed for small values of m to ensure that the higher order terms in the series solution of Rossow would be negligible. A typical comparison of the two solutions is given in Fig. 1 for $m = 0.1$. At each time step of $\Delta t = 0.2$, the local displacement thickness at $x = 1.0$ was determined numerically by discretization of the equation

$$\delta_1 = \int_0^{y_{max}} \left(1 - \frac{u}{u_\infty}\right) dy, \quad (8)$$

and compared to Rossow's result of

$$\delta_r = \frac{m}{u_\infty} y_{max} + \left(1.73 + 0.54 \frac{m}{u_\infty}\right) \frac{1}{\sqrt{Re_\infty}}. \quad (9)$$

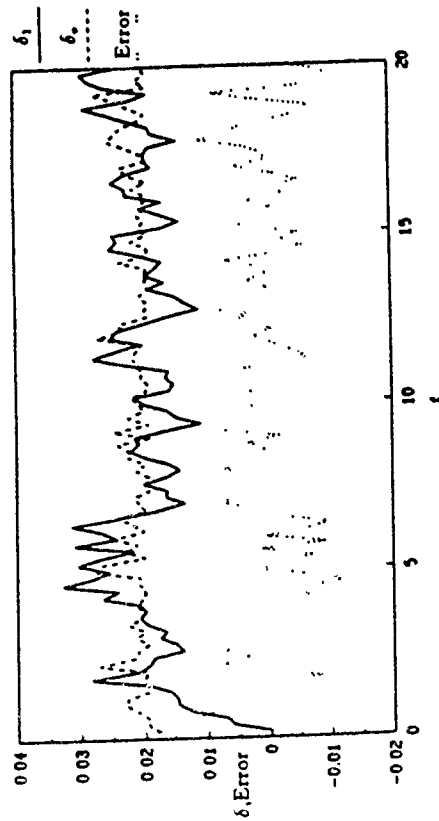


Figure 1 Displacement Thickness.

For calculation purposes, y_{max} was taken as the distance from the plate to the uppermost vortex sheet. The error given in Fig. 1 is determined by $\delta_1 - \delta_1^*$. As shown in Fig. 1, the mean error converges suitably to 0 for $8 \leq \eta \leq 12$.

3. Velocity Profile

The velocity profile was determined by averaging the program results over the 20 times steps for which $8 \leq \eta \leq 12$. To further reduce the stochastic error, the average of 6 runs was obtained for each value of m and is shown in Fig. 2 for values of m ranging from 0 to 0.7 in increments of 0.1. The results shown in Fig. 2 represent the local velocity at $x = 1$, and are presented in terms of the similarity variable $\eta = y/\sqrt{Re_\infty}$.

For $m = 0$ (the hydrodynamic case) the results are essentially the same as those given by Chorin [2]. For $m = 0.1$ and $m = 0.2$, the velocity profile closely resembles that given by Rossow [5] who presents results for $m = 0.1, 0.2, \& 0.5$. The velocity profiles for $m = 0.2$ and $m = 0.3$ are virtually the same. However, for $m > 0.3$, an inflection point develops indicating the tendency for backflow to occur. This tendency becomes more pronounced as m increases until, at $m = 0.6$, backflow occurs. At $m = 0.7$, significant backflow exists for $\eta \leq 2.8$. Rossow's results show the onset of backflow at $m = 0.5$, but he notes that this result is probably not precise as higher order terms may have become notable. Therefore, it appears that application of the vortex method to this problem is

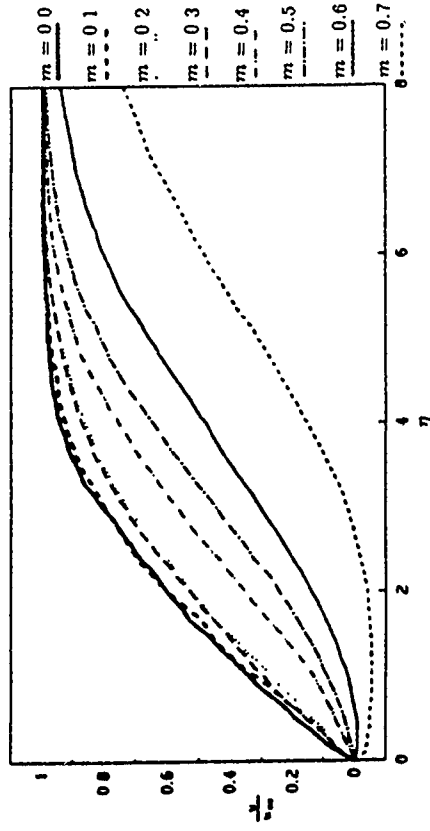


Figure 2. Velocity Profile.

validated by the truncated series similarity solution of Rossow when m remains small, and has overcome the limitations of Rossow's solution for large values of m .

References

- [1] Chorin, A.J. "Numerical Study of Slightly Viscous Flow." *Journal of Fluid Mechanics* 57, pp.785-796, 1973.
- [2] Chorin, A.J. "Vortex Sheet Approximation of Boundary Layers." *Journal of Computational Physics* 27, pp.428-442, 1978.
- [3] Cheer, A.Y. "Numerical Study of Incompressible Slightly Viscous Flow Past Blunt Bodies and Airfoils." *SIAM J. Sci. Stat. Comput.* 4, pp.685-705, 1983.
- [4] Anderson, C.A. "A Vortex Method for Flows with Slight Density Variations." *Journal of Computational Physics* 61, pp.417-444, 1985.
- [5] Rossow, V.J. "On Flow of Electrically Conducting Fluids over a Flat Plate in the Presence of a Transverse Magnetic Field." *NACA Tech. Note* 3971, 1957.
- [6] Cheer, A.Y. "BOUNDL: A Program for Calculating Flow Past a Semi-infinite Flat Plate Using the Vortex Method." LBL 6443, Lawrence Berkeley Laboratory, 1978.

EULERIAN CODES FOR PLASMA SIMULATIONS

M.R FEIX PMMS/CNRS UNIVERSITY OF ORLEANS

Introduction

From the very beginning it has been recognized that Eulerian codes would provide interesting alternative to particle codes. These last ones are "noisy" i.e they exhibit important fluctuations, especially in these phase space regions where the distribution density is very small but where important physical phenomena can take place.

The best and simplest example is the Landau damping phenomena and the subsequent holes formation. Unfortunately the first attempt to treat the Vlasov equation, mostly based on spectral methods (usually Fourier in x and Fourier or Hermite in v) were not very successful for various reasons and Eulerian method were forgotten. Knorr and Cheng gave the correct solution using a splitting methods where the Vlasov equation is solved by advancing a point of the phase space mesh, first in x (keeping the velocity constant), then computing the field and moving along v (at constant x) and then moving again at v constant. So doing we solve Vlasov through the characteristics, which are the trajectories of the particles, and Eulerian codes become very similar to particle ones not only in their spirit but also in the two algorithms. It is very interesting to notice that the treatment of a mesh point (in Eulerian) and a particle require basically the same amount of time (typically 1 μ s per time step on a CRAY II).

Sampling the phase space in Eulerian and particle codes.

The great difference between the two codes is the way phase space is sampled. For Eulerian code we introduce a mesh on the entire phase space and deal with the values of the density distribution which is moved and subsequently redistributed. In particle code a small phase space volume is concentrated on one point where it may be called a "super particle". Usually each super particle corresponds to the same amount of matter and in this respect we deal with an importance sampling scheme which is reminiscent of Monte Carlo method. The comparison is interesting and help to understand when Eulerian or particle codes should be used. If we have sufficient memory space (including the needed speed) to build a phase space mesh of sufficiently small size Eulerian codes must be preferred. If not we must use a particle code, which, as with Monte Carlo algorithms, will be noisy, can completely miss some fine phenomena but gives the behavior of the bulk of the plasma. It is interesting to see where is actually the border:

If we consider a 10^7 mesh points (10s/time step on CRAY II) as a practical limit we see that 3D phase space are on balance (i.e for example, a cylindrical problem with azimuthal symmetry and a sampling of $100(r) \times 300(dr/dt) \times 300(r dr/dt)$). 4D except very special cases, and higher dimensions are still in the domain of particle codes. The

arrival of massively parallel computers will probably promote soon competitive 4D Eulerian codes.

The Interpolation scheme.

After the motion of the mesh point carrying with it the value of f , the obtained distorted mesh must be projected on the Eulerian one. Two methods are possible : The first one use a cubic spline interpolation. It is easy to implement in one dimension, a little bit more complex in higher dimension and gives, usually the best energy conservation test. The second uses successive Fourier transforms on x (conjugate variable k) and on v (conjugate variable λ). The scheme goes as follow :

$$f(x, v, j+1) \Delta t - F \mathcal{F} \mathcal{V} - \tilde{f}(x, \lambda, j+1) \Delta t * \exp i k v \Delta t - I F \mathcal{F} k \\ \tilde{f}(x, v, j+1) \Delta t - F \mathcal{F} \mathcal{V} - \tilde{f}(x, \lambda, j+1) \Delta t * \exp i E \lambda \Delta t - I F \mathcal{F} \lambda - f(x, v, j+1) \Delta t$$

The interest of the Fourier scheme is that it makes the treatment of the Quantum Vlasov equation nearly as simple that the classical one. Indeed the complex structure of the field term in the quantum Liouville equation

$$\frac{i}{2\pi\hbar} \iint \left(\phi(x - \frac{\Delta}{2}, t) - \phi(x + \frac{\Delta}{2}, t) \right) \exp i \frac{(p-p')\Delta}{\hbar} F_w(x, p', \Delta) dp' d\Delta$$

which replaces the term $E \partial f / \partial p$ where $E = -\partial \phi / \partial x$ simplifies considerably when the Fourier transform on p is introduced (we still keep the splitting scheme). In fact in the preceding scheme we just have to replace the phase shift $\exp i \lambda E \Delta t$ which describes the field interaction by the quantity.

$$\exp - \frac{i}{\hbar} \left[\phi(x + \frac{\lambda \hbar}{2}) - \phi(x - \frac{\lambda \hbar}{2}) \right] \Delta t$$

which, indeed, go to the classical limit when $\hbar \rightarrow 0$, and show incidently the nonlocal character of the interaction : L_c is now the quasi Wigner distribution.

Rescaling properties.

An interesting property of the Eulerian codes is the possibility to work in a rescaled time phase space (It is fair to point out that many of these transformations can be extended to particle codes). The idea is that you may have to consider a space domain which varies widely with time (expanding or collapsing plasma for example). In this case the mesh grid and also the time step must be continuously readjusted - an operation numerically difficult. An alternative is the use of a rescaling transformation.

$$x = x C(t) ; dt = A^2(t) d\tilde{t}$$

which imposes since $\dot{\varphi} = d\tilde{t}/d\tilde{t}$

The two scales C and A are at our disposal. By a proper choice we may, sometimes,

$$v = \partial(C/A^2) + \mathcal{L}(dC/dt)$$

guess the asymptotic nature of the solution and ,always, simplify and accelerate the numerical calculations. The case of an expanding two species plasma will illustrate this technique.

Entropy evolution and Eulerian Codes.

A well known property of the Vlasov equation is that it conserves entropy and in fact any quantity of the form :

$$\iint C(f) dx dv$$

Now, if the energy is well conserved with a typical mesh of 200×400 points, entropy is not and increases with time. The reason is the generation of microstructure at smaller and smaller scale (especially in velocity space). When their size reach the phase space mesh size they are smoothed giving a loss of information and an increase of entropy (considered here as a measure of this loss). Two questions are raised and elements of answer are provided :

- Can we keep entropy constant (i.e no loss of information) ?
- Can we loose informations but, nevertheless, trust our result ?

The answer to the first question is that keeping entropy constant is a lost battle. For 1D problem (2D phase space) with 10^7 mesh points we must give up entropy conservation around time $W_{p,1} = 100$ and the problem is not specially wild !

Nevertheless -and this answer the second question in an optimistic way- for reasonably sized mesh, say 10^4 points instead of the 10^7 above mentioned, the lost information never reappears and the jumps of coarse grained entropy do not depend of the size of the mesh and, moreover, agree with the results deduced from the huge (10^7 points) reference simulation where all the details are kept with a very small variation of the fine entropy (as demanded by the mathematical property of the Vlasov equation) and where consequently the increase of entropy comes from our decision of ignoring the microstructure in our measurements (but they are taken into account in the time evolution computation).

Conclusion.

This talk was devoted to the methodology of the Eulerian codes. At the end we will give a list of the treated problems. Now what are our conclusions ?

- Eulerian codes are never "cheap". In the early days, results have been obtained with few hundred particles. The minimum "price" for the 1D is a few 10^4 mesh points.
- With the progress of the supercomputer 1D problem must now always be treated by Eulerian codes. This will be also true pretty soon of 1 1/2D

(3D phase space) problems but work is needed on the different interpolation scheme.

→ With respect to the adaptability of the codes to massively parallel computers Eulerian codes are certainly at least as good as the particle codes (the shifts in the splitting scheme are more systematic but the difference may be small).

→ On the other hand the work on entropy evolution points once again the complex structures of the Vlasov equation. More work is certainly needed (including eventually the influence of a small Fokke - Planck term-which could be very simple since we are interested only in the limit where this term vanish. Such term has, in the past, been helpful in the Fourier Hermite scheme, but this last one was "taylor made" for the Forkker Planck term.

→ Beyond massively parallel computers cellular automata may provide useful device for specialized problems.

Acknowledgements.

First of all we want to thanks the Centre de Calcul Vectoriel pour la Recherche which provides the needed computer time on CRAY I and II.

This paper is entirely based on a large amount of work made by many peoples -PhD students and colleagues - since the very early days of computers simulations. My warmest thanks to all of them.

Finally I want to give the names of P. Bertrand, E. Fijalkow, A.Ghizzo and M. Shoucri. Not only they worked on the different mathematical and physical problems but they keep under constant control the numerous "monsters" called systems, connection networks, mass memories video cassettes, movies etc... always found around the supercomputers.

Bibliography.

- C.Z.Cheng and G.Knorr J.Comput.Phys 22, 330, 1976
- N.D.Suh, M.R.Felix, P. Bertrand J. Comput. Phys 94, 403, 1991
- P. Bertrand and alii Proceeding on non linear Vlasov plasma, Cargese France, 1988
- F.Doveil ed: les Editions de Physique, les Ulis, 1989, p 109
- A. Ghizzo and alii, Phys of fluids 31, 72, 1988.

On Enforcing Conservation Laws in Electromagnetic Particle-in-Cell Codes

A. BRUCE LANGDON

Physics Department, X Division
Lawrence Livermore National Laboratory
University of California
Livermore, CA 94550, USA

We discuss features of possible interest in designing electromagnetic PIC codes with curvilinear coordinates:

- Another view on enforcing Gauss' law - Boris and Marder revisited. A version of Marder's divergence correction gives the same results as an "incomplete Boris" correction in which the exact Poisson solve is replaced by a single point-Jacobi iteration. This observation suggests further improvements.
- We show simple requirements for the existence of algebraically exact conservation laws for the Maxwell equations (for field flux and energy flux) as finally represented in the code. These are found without resort to ad-hoc selection of finite-elements. The symmetries of the finite representation of Maxwell's equations provide stability of the vacuum electromagnetic fields at all time steps below the CFL limit.

Introduction

Correction of $\nabla \cdot \mathbf{E}$

During the 20 years in which two-dimensional electromagnetic particle-in-cell codes have been in intensive use, discussion has continued as to how best to honor both the Ampere-Maxwell equation and Gauss' law in spite of compromises made in the collection of the source terms ρ, \mathbf{J} from the particle coordinates. It is possible to collect a \mathbf{J} whose divergence does correspond to changes in ρ , but at the expense of noise in the solenoidal part of \mathbf{J} . As discussed in [1,2], in the collection of current density \mathbf{J} from the particle coordinates $\{\mathbf{x}, \mathbf{p}\}$, many implementations compromise accuracy in the charge continuity equation $\nabla \cdot \mathbf{J} + \partial \rho / \partial t = 0$ in favor of noise reduction in the solenoidal part of \mathbf{J} , and therefore in the propagating electromagnetic fields. The resulting error in $\nabla \cdot \mathbf{E} = \rho$ is later corrected for.

Boris' popular and very effective procedure erases the error in $\nabla \cdot \mathbf{E}$ by corrections involving solution of a Poisson equation, commonly done by direct methods accurate to machine precision, which may be onerous in complicated meshes.

Marder has recently popularized a partial correction that is very simple and fast, even with complex meshes[1]. Some idiosyncrasies have been reported however; Marder [1] speaks of "excessive" amounts of corrective pseudo-current. No such idiosyncrasies have been observed with corrections based on accurate Poisson solutions, to this author's knowledge.

A new viewpoint on Marder's correction suggests improvements and new possibilities. In this note, I restate Marder's procedure as given in [1], and make a simple modification that improves it and clarifies its relation to Boris' divergence correction. Finally, I show that the improved Marder scheme gives the same results as an "incomplete Boris" correction in which the exact Poisson solve is replaced by a single point-Jacobi iteration. This observation suggests further improvements.

The choice is not limited to, on the one hand, the Boris-type correction involving an accurate solution of Poisson's equation, or, on the other hand, Marder's partial correction.

Conservation Laws and Spatial Differencing of the Electromagnetic Fields

Our BICrz code features a near-orthogonal curvilinear mesh in r, z . Design of the spatial differencing is guided by the conservation laws of electromagnetism such that there exist summation analogues to the electromagnetic conservation laws in integral form that reduce to the correct integrals in the limit of fine orthogonal zoning and are algebraically exact (satisfied to computer precision) for any zoning.

Formal recipes for generating equations with such properties have been advocated, based on finite-element methodologies. The emphasis here is on showing minimal requirements for existence of conservation laws in the equations finally used in the code.

An additional benefit is exact algebraic neutral stability of the vacuum fields.

Correction of $\nabla \cdot \mathbf{E}$

Marder's procedure, and an apparently beneficial modification

In some recent applications, the growing error in $\nabla \cdot \mathbf{E} = \rho$ is reported to be satisfactorily arrested by a very swift diffusion procedure. Almost as described in [1], the Ampere-Maxwell equation is modified:

$$\frac{\mathbf{E}^{n+1} - \mathbf{E}^n}{\Delta t} = c \nabla \times \mathbf{B}^{n+1/2} - \mathbf{J}^{n+1/2} + \nabla F^n, \quad (1)$$

with

$$F^n = d(\nabla \cdot \mathbf{E}^n - \rho^n) \quad (2)$$

(with leap-frog time differencing). This is the same as (10) in [1] with the labeling of time levels shifted half a step.

Marder calls ∇F a pseudo-current. Here the factor d is moved inside the gradient in order to allow d to vary in space while still not disrupting the solenoidal part of \mathbf{J} ; this slight generalization is used when the mesh cells are nonuniform.

Equations (1-2) are using the errors in Gauss' law at time t^n , due to errors in $\nabla \cdot \mathbf{J}^{n-1/2}$ and earlier currents, to correct $\mathbf{J}^{n+1/2}$ and thence improve $\nabla \cdot \mathbf{E}^{n+1}$. However this does nothing to correct additional error in $\nabla \cdot \mathbf{E}^{n+1}$ about to be added due to error in $\nabla \cdot \mathbf{J}^{n+1/2}$ as well. Doing so is easily arranged.

Define

$${}^{(0)}\mathbf{E}^{n+1} = \mathbf{E}^n + c \Delta t \nabla \times \mathbf{B}^{n+1/2} - \Delta t \mathbf{J}^{n+1/2} \quad (3)$$

$${}^{(0)}F^{n+1} = d(\nabla \cdot {}^{(0)}\mathbf{E}^{n+1} - \rho^{n+1})$$

then

$$\mathbf{E}^{n+1} = {}^{(0)}\mathbf{E}^{n+1} + \Delta t \nabla F^n \quad (4)$$

$$= {}^{(0)}\mathbf{E}^{n+1} + \Delta t \nabla d(\nabla \cdot \mathbf{E}^n - \rho^n)$$

is the result of Marder's procedure, (1) above. We may expect to be better off using

$$\mathbf{E}^{n+1} = {}^{(0)}\mathbf{E}^{n+1} + \Delta t \nabla {}^{(0)}F^{n+1} \quad (5)$$

$$= {}^{(0)}\mathbf{E}^{n+1} + \Delta t \nabla d(\nabla \cdot {}^{(0)}\mathbf{E}^{n+1} - \rho^{n+1})$$

so that we adjust for errors resulting from $\nabla \cdot \mathbf{J}^{n+1/2}$ as well as earlier current errors. In the case in which d is uniform, this is the same as Eq. (5) in [3].

I show [6] that F and ${}^{(0)}F$ both evolve according to (5), but the divergence error in \mathbf{E}^{n+1} is smaller in the revised method, for values of d allowed for stable evolution of F , and for the same time sequence of ρ, \mathbf{J} , and therefore of F^n . This indicates that this modification makes the Marder correction more accurate, because a more up-to-date F is used in the correction (5).

* However, in a comparison of self-consistent simulations, since \mathbf{E} differs, the particle ρ, \mathbf{J} will subsequently differ, and therefore also F .

Boris correction with approximate Poisson solution.

The procedure used by Boris, and many others since [2, section 15-6], replaces (4) or (5) by

$$\begin{aligned}\Sigma^{n+1} &= {}^{(0)}\Sigma^{n+1} - \nabla\psi \\ \nabla^2\psi &= \nabla \cdot {}^{(0)}\Sigma^{n+1} - \rho^{n+1}\end{aligned}\quad (6)$$

As usually described and implemented, the Poisson solution is by a direct method to machine precision.

If instead we just do accelerated point Jacobi,

$$\begin{aligned}\psi_{j,a}^{(m+1)} - (1-\omega)\psi_{j,a}^{(m)} &/\omega \\ &= \psi_{j-1,a}^{(m)} + \psi_{j+1,a}^{(m)} + \psi_{j,a-1}^{(m)} + \psi_{j,a+1}^{(m)} - \Delta x^2 \left(\nabla \cdot {}^{(0)}\Sigma^{n+1} - \rho^{n+1} \right)\end{aligned}\quad (7)$$

c.) a square mesh in two dimensions, with $\psi^{(0)} = 0$, we find $\psi^{(1)}$ is the same as ${}^{(0)}F^{n+1}$ in (5), when $d\Delta t = \omega\Delta x^2/4$. Thus the improved Marder correction is the same as a Boris correction in which the Poisson solution is approximated by a single pass of point Jacobi iteration.

There is a range of options, more effective than this, but short of an exact Poisson solution that completely corrects $\nabla \cdot \mathbf{E}$. These can be drawn from the lore of iterative solution of Poisson's equation, keeping in mind that the spectrum of the source term in (6) is dominantly at short wavelengths. One possibility is to apply Jacobi iteration more than once. A related iteration is used in [3]. The AVANTI code has always used a few alternating-direction-implicit iterations to do an implicit-PIC form of incomplete Boris correction [4].

Note however that time reversibility ([2], p. 361) may be lost when the correction is incomplete. We are reminded [5] that continuity error at the boundary is also important. Prolonged continuity errors of one sign, such as when there is a net particle current through a boundary, would cause nonzero F to grow and diffuse into the interior [5]. Consistent formulation of wall charge and $\nabla \cdot \mathbf{E}$ at the boundary, as also done with the Boris correction [2, section 15-12], averts such problems.

Conservation Laws and Spatial Differencing of the Electromagnetic Fields

The spatial difference forms for Maxwell's equations in BICrz's ASO coordinates, leaving time as a continuous variable for now, are of the form

$$\left[(h_1\Delta x_1 h_2\Delta x_2) \frac{\partial B_z}{\partial t} \right]_{j+\frac{1}{2},k+\frac{1}{2}} = (E_1 h_1 \Delta x_1)_{j+\frac{1}{2},k+\frac{1}{2}} - (E_1 h_1 \Delta x_1)_{j-\frac{1}{2},k+\frac{1}{2}} - (E_2 h_2 \Delta x_2)_{j+\frac{1}{2},k+\frac{1}{2}} + (E_2 h_2 \Delta x_2)_{j-\frac{1}{2},k+\frac{1}{2}} \quad (8)$$

$$\left[(h_2\Delta x_2) \frac{1}{c^2} \frac{\partial E_1}{\partial t} \right]_{j,k+\frac{1}{2}} = (rB_z)_{j+\frac{1}{2},k+\frac{1}{2}} - (rB_z)_{j-\frac{1}{2},k+\frac{1}{2}} - \frac{\mu_0}{2\pi} I_{j+\frac{1}{2},k+\frac{1}{2}} \quad (9)$$

etc

$$\begin{aligned}\frac{1}{2\pi\epsilon_0} Q_{j,a} &= (E_1 r h_2 \Delta x_2)_{j+\frac{1}{2},a} - (E_1 r h_2 \Delta x_2)_{j-\frac{1}{2},a} \\ &\quad + (E_2 r h_1 \Delta x_1)_{j,a+\frac{1}{2}} - (E_2 r h_1 \Delta x_1)_{j,a-\frac{1}{2}}\end{aligned}\quad (10)$$

Q and I are charge associated with a mesh point, and current through cell sides (not densities), and are collected as such from the particles. The continuity equation is

$$\frac{\partial}{\partial t} Q_{j,a} = -I_{j+\frac{1}{2},a} + I_{j-\frac{1}{2},a} - I_{j,a+\frac{1}{2}} + I_{j,a-\frac{1}{2}} \quad (11)$$

The conservation laws are a consequence simply of the form of these equations, in which the same expressions are used for, e.g., $\rho h_1 \Delta x_1$ in $\nabla \cdot \mathbf{E}$, $\nabla \times \mathbf{B}$, and in the $\epsilon_0 E^2/2$ energy.

The code requires only that values be chosen for a mapping to positions $z_{j,a}$, $r_{j,a}$ of the cell indices j,a . Accuracy (but not the conservation laws) requires that this mapping approximate an orthogonal coordinate transformation.

The field flux conservation laws and the algebraically-exact Poynting theorem is too long to reproduce in the space available here.

BICrz's differencing on a near-orthogonal mesh provides provably-stable electromagnetic oscillations.* The dispersion relation for vacuum electromagnetic cavity modes is of the form

$$-(\nabla \times \nabla \times B_z \phi)_\omega - \left(\frac{2}{\Delta t} \sin \frac{\omega \Delta t}{2} \right)^2 B_z = 0 \quad (12)$$

The difference operator $-(\nabla \times \nabla \times B_z \phi)_\omega$ is symmetric and positive-definite. Therefore, the value of $[(2/\Delta t) \sin(\omega \Delta t/2)]^2$ is real and is positive for each eigenmode. We have neutral stability for any time step Δt below the threshold for Courant instability.

Conclusions

A very modest approximation to a Poisson solution in Boris's correction will be as successful as Marder's correction. Understanding of when and how partial divergence corrections succeed and fail would be of interest.

Ensuring the existence of finite analogues to the conservation laws of EM is not hard to arrange, given fields defined on interleaved meshes and given near-orthogonality of the cells.

Acknowledgments

Helpful discussions were had with J.-C. Adam, D. W. Hewett, B. Marder, and D. Nielsen, Jr. This work was performed under the auspices of the U. S. Department of Energy by the Lawrence Livermore National Laboratory under Contract No. W-7405-Eng-48.

References

1. B. MARDER, "A Method for Incorporating Gauss' Law into Electromagnetic PIC Codes", *J. Comput. Phys.* **68**, 48 (1987).
2. C. K. BIRDSALL AND A. B. LANGDON, *Plasma Physics Via Computer Simulation*, McGraw-Hill, New York, 1985.
3. D. E. NIELSEN, JR. AND A. T. DROBOT, "An Analysis and Optimization of the Pseudocurrent Method", *J. Comput. Phys.* **89**, 31 (1990).
4. D. W. HEWETT AND A. B. LANGDON, "Electromagnetic Direct Implicit Plasma Simulation", *J. Comput. Phys.* **72**, 121-155 (1987).
5. B. MARDER, to be published.
6. A. B. LANGDON, "On Enforcing Gauss' Law in Electromagnetic Particle-in-Cell Codes", UCRL-JC-105059, to be published.

* Not true for some other EM codes providing more general quadrilateral meshes

TWO AND THREE DIMENSIONAL PARTICLE-IN-CELL SIMULATION WITH MAFIA

Petra Schuett and Thomas Weiland
Technische Hochschule
Fachbereich 18
6100 Darmstadt, GERMANY

ABSTRACT

The MAFIA 3D family of computer codes has been well established in the frequency and time domain, and has proven its reliability by comparison with measured results in many laboratories around the world. With the entirely overhauled Version 3.X, which was first released in April 1990, MAFIA now also contains 2.5D and full 3D particle-in-cell (PIC) modules. As MAFIA also has modules for electrostatic fields, (nonlinear) magnetostatic fields, RF fields, low-frequency fields, any combination of external fields may be used consistently as starting fields for a fully electromagnetic PIC simulation. For example, for klystron modeling, one may precalculate the high voltage gun field, the (nonlinear) magnetic focusing field, and the RF modes in the output cavities, etc. All such precalculated fields may be easily preloaded, including scaling for amplitudes and phase. The new PIC modules in MAFIA 3.1, namely TS2 and TS3, are menu driven with context-sensitive on-line help. Thus they are well suited for modern workstations and enable a fully electromagnetic analysis of complex structures, such as klystrons, crossed-field amplifiers, etc.

We will illustrate the talk with examples of different types of MAFIA simulations, emphasizing PIC applications.

Simulation of High-Current Ion Beams in General Accelerator Lattices*

Alex Friedman and David P. Grote

Lawrence Livermore National Laboratory, Livermore CA 94550

Irving Haber

U.S. Naval Research Laboratory, Washington DC 20375

The beams in a Heavy Ion beam driven Inertial Fusion (HIF) accelerator are collisionless, non-neutral plasmas, confined by applied magnetic and electric fields. These "space charge dominated" beams must be focused onto small (few mm) spots at the fusion target, and so preservation of a small "emittance" (roughly, phase-space volume) is crucial.¹ The nonlinear beam self-fields can lead to emittance growth, especially when the beam undergoes many manipulations, including transport through bends and (possibly misaligned) focusing elements, non-steady acceleration, injection into rings, etc. Thus, a self-consistent field description is needed; particle-in-cell (PIC) techniques have been shown to be particularly well suited.²

We have developed a simulation program, WARP, to study these beams. The code's 3d package, known as WARP6 because of its "warped Cartesian" geometry and 6d phase space, combines features of an accelerator code and a PIC plasma simulation. Novel techniques, motivated by the need to follow beams through many accelerator elements over long distances and around bends, have been developed and are described here. Using these methods, we have followed beams around single bends and around multiple laps of a "racetrack" system. The code has also been applied to a number of other problems, described in the references.

CODE OVERVIEW

The WARP code consists of: WARP6 (described here and in 3-7); WARPZ,³ an r, z (axisymmetric) PIC package; an envelope equation solver (used for loading a near-equilibrium "matched" beam, and for lattice "tuning"); and facilities for diagnostics, etc. The code uses BASIS,⁴ which provides a development system and an interactive user interface.

The code's accelerator "lattice" consists of a fully general set of finite-length (for now, sharp-edged) focusing and bending elements. E and $v \times B$ forces from these elements are computed in the local laboratory frame algebraically at each particle location. For efficiency, we load a uniform 1d grid with lattice information (element starts, ends, strengths, etc.) at the beginning of each timestep. When advancing the particles we extract data from this grid, rather than from the master lists of elements. Axial confinement, obtained in real experiments by shaping the accelerating pulses as the beam ends pass by, is currently modeled by applying axial fields which move with the beam.⁵ The computational mesh for the self-field, which is assumed electrostatic in the beam frame, moves with the beam and is laid down anew at each timestep. To model mildly-relativistic driver-scale beams, we plan to obtain the lab-frame self- E and B via Lorentz transformations (at least in simple straight systems), but for now we use E_{self} directly.

The self-potential ϕ is obtained via vectorized sine-sine-periodic FFT operations "in place." Self-field boundary conditions are most naturally those of a square metal pipe in x, y . A round pipe, or other z -independent shape, can be obtained by use of a 2d (transverse) capacity matrix applied independently to each axial Fourier mode.⁶ More general methods are under development.⁷

No mesh arrays for the components of E are used; instead, ϕ is gathered from 32 cells in the neighborhood of each particle, and then differentiated. This saves the space of three 3d arrays.

In leapfrog motion, if a particle were to land within a sharp-edged focusing or bending element on four steps while its neighbor did so on only three, they would receive dramatically different impulses. Thus, the advance is modified to incorporate "residence corrections" for element forces;

these multiply the applied field by the fraction of the velocity advance step spent within the element. This allows much bigger steps than are otherwise possible.

The particle advance is vectorized. Deposition of the charge density ρ is vectorized with length 8, over cells overlapped by each particle.

Elongated zones (aspect ratios of order 10:1) have been found to work well provided the axial zone size is less than the beam radius. The timestep size Δt is chosen to resolve external field gradients; accurate trajectories are necessary. The plasma period is long ($\sim 100\Delta t$) and well-resolved.

BENT-BEAM PARTICLE DYNAMICS

We have developed a family of techniques for modeling bends. These are based upon following a particle's position and velocity in a sequence of Cartesian (laboratory) frames. This "warped" coordinate system is natural for the description of accelerators which include bends.

In the "exact" method presented previously,¹⁰ a particle is advanced using (e.g.) a leapfrog advance in the coordinate system associated with the particle's location at the beginning of the step. Then, x and v are algebraically transformed into the rotated inertial coordinate system associated with the particle's location at the end of the step. No pseudo-forces are used. The scheme conserves phase-space area identically, and implies no large-aspect-ratio (gentle bend) expansion. The numerical properties are just those of the underlying difference scheme. The method has a nontrivial operation count, but is quite usable. It has been successfully applied in the ELBA electron accelerator code.¹¹

Here, we concentrate upon the "simplified" method now used in WARP6. The method is inexact, but is faster and accurate enough for present purposes. "Intermediate" methods are also possible. In our coordinate system, the radius of curvature of the reference orbit (usually the vessel centerline) is r_c . Time is the independent variable for particle orbits. The conventional (for accelerator codes) independent variable s is in WARP a dependent variable for orbits, as are the transverse coordinates x, y . In the straight sections, $s \approx z$, while in the bends, $s \approx -r_c\theta$. The "radial" coordinate is $x \approx r - r_c$; the unit vectors \hat{x} and \hat{y} evolve as a particle moves, and are different for each particle. This geometry is depicted in Fig. 1(a). The axial speed is $v_z = -r\dot{\theta}$. The axial position (projection onto the centerline) is advanced in time using:

$$ds/dt = -r_c\dot{\theta} = (r_c/r)v_z. \quad (1)$$

In this derivation of the simplified bent-beam algorithm, we show that the necessary "pseudo-forces" are equivalent to a velocity rotation. The radial coordinate increases for an unaccelerated particle due to "centrifugal force," with $q/m = 1$:

$$\frac{d}{dt}v_z = \frac{v_z^2}{r_c + z} + \{E + v \times B\}_z. \quad (2)$$

For the "Coriolis force," use conservation of angular momentum:

$$\frac{d}{dt}r^2\dot{\theta} = r\{E + v \times B\}_\theta. \quad (3)$$

After algebra:

$$\frac{d}{dt}v_z = -\frac{v_z v_r}{r_c + z} + \{E + v \times B\}_z. \quad (4)$$

Note there is no factor of two in the Coriolis force because the frame of reference differs from that in the classical analysis; particles velocities retain their laboratory-frame magnitudes. Considering only the "pseudo-force" terms, the rate of change of the velocity angle is:

$$\frac{d}{dt} \arctan \left(\frac{v_y}{v_x} \right) = \frac{v_r}{r_c + z}. \quad (5)$$

The transformation is a pure rotation; one can readily verify that $d|v|/dt = 0$.

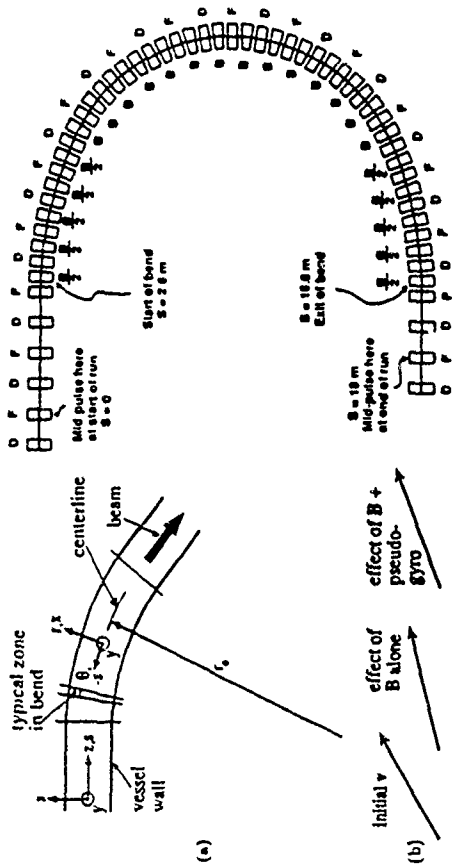


Figure 1. (a) Geometry. (b) Effect of advance. Implementation in WARP was straightforward. We needed only to augment the physical dipole (bending) field with a "pseudo-gyrofrequency":

$$B_{p,dipole} \leftarrow B_{p,dipole} - \frac{m}{q} \frac{v_r}{r_c + z} \quad (7)$$

This folds the necessary back-rotation into existing coding; residence corrections must be used on bend entry or exit. The net effect is depicted in Fig. 1(b). We must advance the axial position z using Eq. (1), also residence-corrected. The algorithm is inexact because v_r and z are non-constant.

BENT-BEAM SELF-FIELD

We solve Poisson's equation in "warped" coordinates using a simple, rapid iteration. Poisson's equation (using $h \equiv 1/r_c$) is:

$$\frac{1}{1 + hz} \frac{\partial^2 \phi}{\partial z^2} + \frac{1}{1 + hz} \frac{\partial}{\partial z} \left(\frac{1}{1 + hz} \frac{\partial \phi}{\partial z} \right) = -4\pi \rho \quad (8)$$

We differentiate the products, and move all terms except the dominant "Cartesian" second derivative terms to the right member. The latest ϕ is used to explicitly compute the smaller "non-Cartesian" terms, updating the "source." At each iteration a 3d FFT inverts the Cartesian ∇^2 operator. The iteration converges rapidly, to relative changes in ϕ of 10^{-6} in two or three passes; this corresponds to a typical relative error in the Poisson equation of $\sim 10^{-8}$.

In a bend, axial zone spacing varies with z ; thus we use $\rho = \rho_c r_c / r$, where ρ_c is collected from the particles as for a straight system. The axial field is $E_z = -(r_c/r) \partial \phi / \partial z$.

ILLUSTRATIVE RESULTS

A typical lattice of interest to the upcoming ILSE experiments¹² is shown in Fig. 2; the F 's and D 's denote focusing and defocusing quadrupole magnets for the in-plane motion, the B 's and $B/2$'s are full- and half-strength bending magnets.

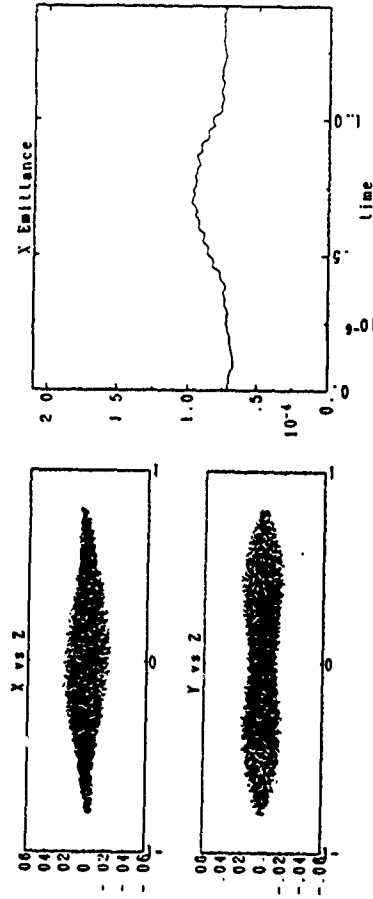


Figure 3. Beam after 600 steps.

Figure 4. Mid-pulse z emittance vs. time.

Fig. 3 shows the particle distribution of an axially-hot beam after the center of the beam has moved 12 m (600 steps); the upper half is a "top view" of the beam, while the lower half is a "side view." The radially inward displacement of the beam tail (toward negative z) is a consequence of the tail's reduced axial velocity, which results from the axial expansion. Fig. 4 shows a time-history of the in-plane transverse emittance at mid-pulse. The hump at mid-pulse is a result of dispersion; while it diminishes upon bend exit, some residual emittance growth is present. This run used a $64 \times 64 \times 128$ mesh and 54160 particles. It took approximately two hours on a Cray X/MP48.

Work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract W-7405-ENG-48, and by the U.S. Naval Research Laboratory under Lawrence Berkeley Laboratory contract DE-AC03-76SF0098.

1. T. J. Fessenden and A. Friedman, "Heavy Ion Inertial Fusion," LLNL Report UCRL-JC-106789, 1991; Nucl. Fusion, in press.
2. I. Haber, "High-Current Simulation Codes," in AIP Conf. Proc. 139 (AIP, New York, 1986).
3. A. Friedman, D. A. Callahan, D. P. Grote, A. B. Langdon, and I. Haber, "WARP: A 3D (+) PIC Code for HIF Simulations," Proc. Conf. on Comp. Codes and the Linear Accel. Community, Los Alamos, January 21-25, 1990. LANL Report LA-11857-C, 119-136, 1990.
4. A. Friedman, R. O. Bangerter, D. A. Callahan, D. P. Grote, A. B. Langdon, and I. Haber, "A 3D Particle Simulation Code for Heavy Ion Fusion Accelerator Studies," Proc. 2nd European Part. Accel. Conf., Nice, June 12-16, 1990.
5. A. Friedman, D. P. Grote, D. A. Callahan, A. B. Langdon, and I. Haber, "3D Particle Simulation of Beams Using the WARP Code: Transport Around Bends," LLNL Report UCRL-JC-104981 Rev. 1, 1991; to appear in Part. Accel.
6. D. P. Grote, A. Friedman, and I. Haber, "3D Simulations of Axially Confined Heavy Ion Beams in Round and Square Pipes," LLNL Report UCRL-JC-104408, 1991; to appear in Part. Accel.
7. D. P. Grote et al., this conference.
8. D. A. Callahan et al., this conference.
9. P. F. Dubois, et al., "The Basic System," LLNL Document M-225 (1988).
10. A. Friedman, "Methods for PIC Simulation of Bent Particle Beams in 3d and 2d," paper PM1-10, Proc. 13th Conf. on Numerical Sim. of Plasmas, Santa Fe, 1989.
11. G. Joyce, J. Kral, and S. Slinker, "ELBA-A Three-Dimensional Particle Simulation Code," Proc. Conf. on Comp. Codes and the Linear Accel. Community, Los Alamos, January 21-25, 1990. LANL Report LA-11857-C, 98-102, 1990.
12. E. P. Lee, Nucl. Inst. Meth. Plasma Res. A278, 178 (1989).

The ARGUS Code

John J. Peullo, Alan Mankofsky, Adam T. Drobo
Allred A. Mondelli, Warren A. Krueger, and Michael E. Kress*

Science Applications International Corporation
1710 Goodridge Drive
McLean, VA 22102

*also at
City University of New York
College of Staten Island
130 Guyanessan Place
Staten Island, NY 10310

ARGUS Code Philosophy

The ARGUS code¹ has been under development at SLAC since 1983. It is a general-purpose three-dimensional simulation code. The code architecture is specifically designed to handle the problems associated with three-dimensional simulations, and was not developed by reworking an older two-dimensional code. It uses sophisticated memory management and data handling techniques to deal with the large volume of data that is generated in three-dimensional simulations.

A modular architecture is employed so that ARGUS is in fact a system of three-dimensional codes (numerical modules) that utilize a common data structure and share utilities for structure input, grid generation, memory management, data handling, and diagnostics. The codes allow complicated geometrical structures to be represented on the computational grid. The grid can be nonuniform in all three dimensions. Physics modules are in place to compute electrostatic and electromagnetic fields, the eigenmodes of rf structures, and particle motion in either a time-dependent mode or a steady-state mode. The particle-in-cell (PIC) modules include multiple species, the Lorentz equations of motion, phenomenological terms for elastic and inelastic scattering, and algorithms for the creation of particles by emission from material surfaces, injection onto the grid, and ionization.

Main Modules of ARGUS

Domain Decomposition

ARGUS uses a domain-decomposition algorithm,² coupled with memory-management and data-handling techniques, to optimize the use of core memory for each problem and to efficiently move data between core and disk memory during the calculation. (The advent of solid-state data has reduced disk access times to the point where massive disk I/O no longer constitutes an unacceptable overhead.) A large problem is divided into blocks which are independently processed in core. The data-handling module uses overlapped I/O to move the data blocks between disk and core as they are needed by various code modules. The ARGUS physics modules are designed for compatibility with this data structure. The disk I/O is able driven, so that the disk I/O sequence for each algorithm can be independently optimized. With these techniques the maximum problem size in ARGUS is limited by the available disk storage and by the overhead of disk I/O, rather than by the size of core memory. This architecture is also compatible with eventual parallel processing within ARGUS; each block could be processed by a separate CPU. The data-handling module attempts to minimize disk I/O by using available data in core whenever possible. Small problems usually remain core-resident.

Structure Specification via Combinatorial Geometry

The structure input in ARGUS is carried out through combinatorial geometry. The code stores a library of basic three-dimensional objects (e.g., a rectangular solid, an elliptical cylinder, an ellipsoid, a paraboloid, etc.). These objects are combined by the user with logical operations (to either add or delete the library object) to produce structures of arbitrary shape. The structures so specified are represented on the computational grid by a structure mask array, which stores the material and electrical properties of each cell on the grid.

Material properties can be associated with structures. The code allows perfectly-conducting materials, as well as materials with complex values of both permittivity and permeability; hence lossy materials are allowed. Furthermore, the permittivity and permeability may be specified as diagonal tensors to treat certain classes of non-isotropic materials.

Field Solvers

The three-dimensional fields required in ARGUS can be computed either in the electrostatic approximation or as solutions to the full time-dependent Maxwell's equations. Two types of three-dimensional electrostatic solvers (using direct or iterative methods) and two types of three-dimensional electromagnetic solvers (frequency domain or time domain) are available within ARGUS.

The options for electrostatic solutions consist of a general, iterative solver for Poisson's equation, using Chebyshev-accelerated SOR, or a direct Fast-Fourier-Transform method with a sparse-matrix technique to match the field across field blocks and to satisfy the boundary conditions on internal structures embedded on the grid.

Alternatively, ARGUS can solve the complete set of Maxwell's equations in either the time domain or the frequency domain. The solutions for the electric and magnetic fields can be analyzed by the code to provide integrated field quantities, such as impedances, field energy, etc. The frequency-domain algorithm³ solves the electromagnetic eigenvalue problem for the normal modes of an rf structure. The time-domain algorithm solves the initial-value problem using an explicit "leap-frog" integration technique. The initialization of the fields can be carried out by introducing waves at ports, by driving specified currents on the grid, or by initializing specific fields on the grid. The electric and magnetic fields can be recorded as functions of time at specified witness points and Fourier analyzed to obtain the frequency content of the solution.

Particle-In-Cell (PIC) Simulation

PIC simulation offers a means of solving for charged-particle flows and fields in a self-consistent manner. As the charged particles move in the local electric and magnetic fields, they alter the local space charge and current density fields. These new source fields then act in Maxwell's equations to modify the electric and magnetic fields. The PIC algorithm couples these two phenomena on every time step in time-dependent PIC, or on each iteration in steady-state PIC simulations.

In the steady-state mode, ARGUS iteratively solves for the self-consistent fields and particle flows in the manner of a "gun code". The full time-dependent PIC algorithm solves the particles flows and fields self-consistently within each time step of the simulation. This algorithm allows transients to be studied, and can be used to analyze unstable interactions between particles and between particles and fields.

Diagnostics

ARGUS has a wide variety of diagnostic plots that are selected by user input and are available at run time. These include (1) setp diagnostics that display the grid and structures, (2) field visualizations as contour plots and arrow plots as well as time histories of field components at selected sample points, and (3) particle diagnostics, consisting of phase space projections on any two-dimensional subsurface of the six-dimensional phase space.

ARGUS Applications

Full PIC simulations have been carried out to study instabilities in the particle flow in a magnetically-insulated transmission line (using electromagnetic fields) and to model heavy-ion beam transport (with electrostatic fields) in a periodic magnetic quadrupole lattice for the heavy-ion fusion program.

Additionally, steady-state PIC simulations have been carried out to model electron and ion sources and depressed electron collectors. Recently, ARGUS has been applied to the design of the helical electrostatic quadrupole low-energy beam transport for the SSC accelerator.

The electromagnetic modules in ARGUS have been used to carry out cold-test calculations for microwave coupled-cavity tubes and accelerator structures. Calculations with lossy materials have been carried out to model severe instabilities in a microwave amplifier and high-order-mode dampers in a gyrotron amplifier circuit. ARGUS has modeled the double output cavity for the SLAC klystron. It has also been applied to the analysis of the rf modes in MMIC circuits and MMIC packaging structures.

Full PIC simulations have been carried out to study instabilities in the particle flow in a magnetically-insulated transmission line (using electromagnetic fields) and to model heavy-ion beam transport (with electrostatic fields) in a periodic magnetic quadrupole lattice for the heavy-ion fusion program.

Additionally, steady-state PIC simulations have been carried out to model electron and ion sources and depressed electron collectors. Recently, ARGUS has been applied to the design of the helical electrostatic quadrupole low-energy beam transport for the SSC accelerator.

Full PIC simulations have been carried out to study instabilities in the particle flow in a magnetically-insulated transmission line (using electromagnetic fields) and to model heavy-ion beam transport (with electrostatic fields) in a periodic magnetic quadrupole lattice for the heavy-ion fusion program.

As a recent, but typical, example of the use of ARGUS, a series of electrostatic time-domain PIC simulations have been carried out on an electron gun being designed and built by Princeton to be used to study spacecraft charging on Space Shuttle missions. Figure 1 shows a 2D representation of the diode configuration from an EGUN simulation and Fig. 1b shows the 3D cathode simulated with ARGUS with its odd-shaped emission surface. In ARGUS, the gun is represented on a mesh and the gun structure resolved on the mesh is shown in Fig. 2. In the simulation, the voltage of the control anode (first electrode) was exponentially increased from zero to simulate the turn-on behavior of the diode. Snapshots of the particle positions in time during this process are shown in Figs. 3 and 4. It can be seen in Fig. 3 that the cathode emission surface shape is somewhat retained all along the beamline. The small rotation was due to the applied external magnetic field. Additionally, the longitudinal views (Fig. 4) show the detailed beam shape, also showing an effect due to the external magnetic field. Experimental results thus far have shown reasonable agreement with the ARGUS results. This study has just recently begun, and ARGUS will continue to be used to study detailed electromagnetic effects as well as equilibrium behavior of the system.

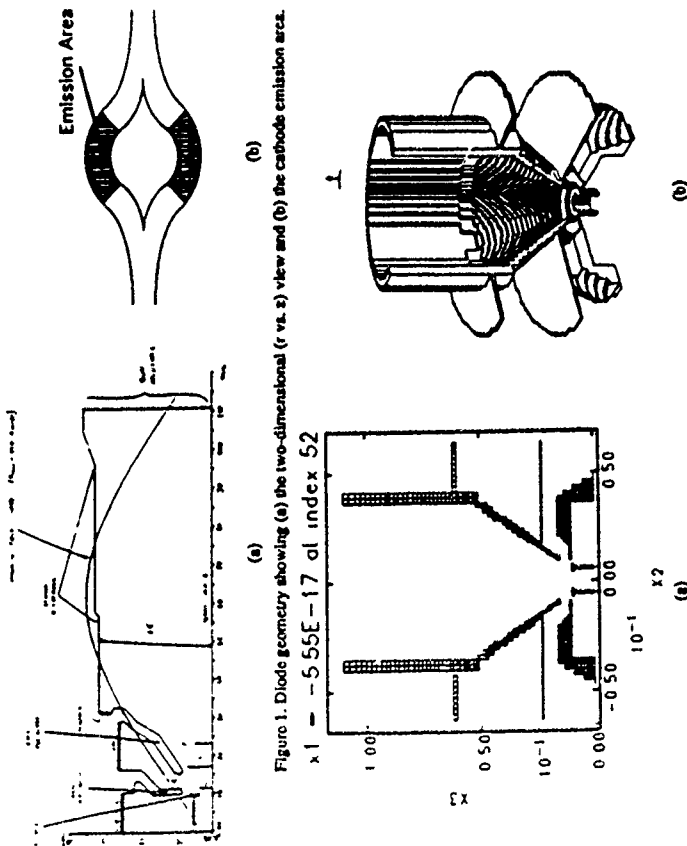


Figure 1. Diode geometry showing (a) the two-dimensional (r vs. z) view and (b) the cathode emission area.

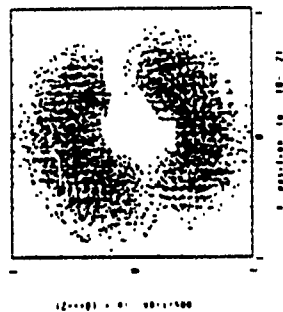


Figure 3. Transverse x vs. y particle plot

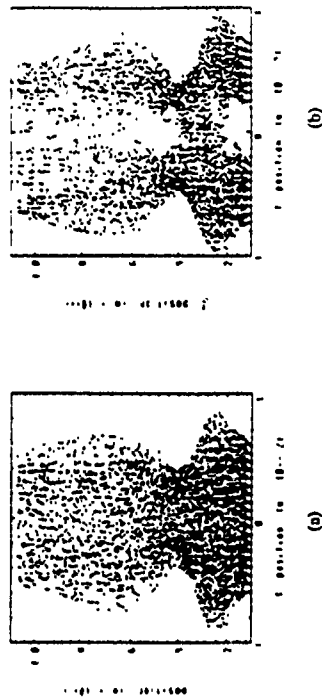


Figure 4. Longitudinal (a) x vs. z and (b) y vs. z particle plots

References

1. A. Mankofsky, *Three-Dimensional Electromagnetic Particle Codes and Applications to Accelerators, Linear Accelerators and Beam Optics Codes*, C.R. Emlinheiser, ed., A.I.P. Conf. Proc. No. 177 (American Institute of Physics, New York, 1988), p. 137ff.
2. A. Mankofsky, J.L. Sefor, C.L. Chang, K. Ko, A.A. Mondelli, A.T. Drobat, J. Mrura, W. Alimurtti, S.T. Brandon, D.E. Nielsen, Jr., and K.M. Dyer, *Comp. Phys. Commun.* **48**, 155 (1988).
3. J. Tuckman, *An Improved Version of the Eigenvalue Processor SAP Applied in URMEL*, CERN/INF 85-4 (1985).

Figure 2. (a) 2D and (b) 3D representation of the diode on the ARGUS grid.

Large Space-Scale and Low Frequency Electromagnetic Particle Simulation of Inhomogeneous Plasmas in Three-Dimensions

Motohiko Tanaka, S Murakami, H Takamaru, and T Sato
National Institute for Fusion Science, Nagoya 464-01, Japan

Abstract

An electromagnetic, implicit particle simulation method in three-dimensions is presented that can be applied to inhomogeneous and finite beta plasmas with large space and time scales. In order to include the diamagnetic effects for strongly magnetized ($\omega_{ce} \geq \omega_{pe}$) and inhomogeneous plasmas, full-implicit coupled-field-particle equations are derived by using the backward time-decentered scheme and the guiding center approximation for the electron perpendicular motion. In this paper, the basic algorithm of the code is described and several physics applications are shown for 1) the Alfvén-ion-cyclotron instability and 2) kink instability of the peaked density ion beam.

1 Introduction

As many people have already noticed, there exists a giant gap in space and time between the microscopic (kinetic) and macroscopic (magnetohydrodynamic) natures of plasmas. Nevertheless, many important and interesting problems of fusion and space plasmas, like anomalous transport, fall on this category. Extensive efforts were made in the 1980's in the United States and Japan to develop new simulation tools to overcome this difficulty.¹⁻³⁾

The macroscale particle simulation code³⁾ (MACROS) developed by the author since early 1984 is based on the insight that the backward time-decentered electromagnetic field acts to selectively suppress high frequency oscillations of field and plasma particles. The macroscale particle simulation code that can deal with high magnetic field ($\omega_{ce} \sim \omega_{pe}$), inhomogeneous plasmas in large space and time scales, i.e., $\omega_{pe}\Delta t \gg 1$ and $\Delta x \geq c/\omega_{pe} \gg \lambda_{De}$, was completed in 1988. The validity of the code was extensively studied and verified both analytically and numerically.³⁾ Recently the code has been upgraded to deal with inhomogeneous, kinetic and high beta plasmas in large (MHD) scales. The characteristic features of the macroscale particle code thus developed are summarized in Table 1.

2. General Algorithm

2.1 Field and particle equations

The Maxwell equations used in the macroscale particle simulation are given by

$$\frac{1}{c} \left(\frac{\partial \mathbf{E}}{\partial t} \right)^{n+1/2} = \nabla \times \mathbf{B}^{n+0} - \frac{4\pi}{c} \mathbf{j}^{n+1/2}, \quad (1)$$

$$\frac{1}{c} \left(\frac{\partial \mathbf{B}}{\partial t} \right)^{n+1/2} = -\nabla \times \mathbf{E}^{n+0}, \quad (2)$$

$$\nabla \cdot \mathbf{E}^{n+1} = 4\pi \rho^{n+1}, \quad (3)$$

$$\nabla \cdot \mathbf{B}^{n+1} = 0, \quad (4)$$

where $\frac{1}{2} < \alpha, \gamma \leq 1$. The equations of motion for the ions are given by

$$\frac{d\mathbf{v}_i^{n+1/2}}{dt} = \frac{e_i}{m_i} [\mathbf{E}^{n+0}(\mathbf{x}_i) + \frac{\mathbf{v}_i^{n+1/2}}{c} \times \mathbf{B}^{n+0}(\mathbf{x}_i)], \quad (5)$$

$$\frac{d\mathbf{x}_i^{n+1/2}}{dt} = \mathbf{v}_i^{n+1/2}, \quad (6)$$

and those for the electron are,

$$\frac{d\mathbf{v}_e^{n+1/2}}{dt} = \frac{(-e)}{m_e} \mathbf{E}_\parallel^{n+0}(\mathbf{x}_e) - \frac{\mu_B}{m_e} \nabla_\parallel \mathbf{B}^{n+0}, \quad (7)$$

$$\mathbf{v}_e^{n+1/2} = c \left(\frac{\mathbf{E} \times \mathbf{B}}{B^2} \right)^{n+1/2} + \left(\frac{m_e c}{-eB} \right) \mathbf{b} \times \left\{ \frac{\mu_B}{m_e} \nabla B + v_\parallel^2 (\mathbf{b} \cdot \nabla) \mathbf{b} \right\}^{n+1/2}, \quad (8)$$

$$\frac{d\mathbf{x}_e^{n+1/2}}{dt} = (\mathbf{v}_\parallel^{n+1/2} + \mathbf{v}_\perp^{n+1/2}), \quad (9)$$

where $\mathbf{b} = (\mathbf{B}/B)$, and the parallel velocity vector is $\mathbf{v}_\parallel^{n+1/2} = v_\parallel^{n+1/2} \mathbf{b}^{n+1/2}(\mathbf{x}_i)$.

2.2 Directly-coupled field-particle equations

In the finite-difference form the Maxwell equations are written as

$$\mathbf{E}^{n+1} - \mathbf{E}^n = c\Delta t \nabla \times \mathbf{B}^{n+0} - 4\pi\Delta t \mathbf{j}^{n+1/2}, \quad (10)$$

$$\mathbf{B}^{n+1} - \mathbf{B}^n = -c\Delta t \nabla \times \mathbf{E}^{n+0}. \quad (11)$$

In order to avoid the Courant condition on the time step Δt , we eliminate \mathbf{B}^{n+1} using the interpolation. $\mathbf{E}^{n+0} = \alpha \mathbf{E}^{n+1} + (1-\alpha) \mathbf{E}^n$:

$$\left[1 + (\alpha c \Delta t / c) \nabla \times \nabla \times \right] \mathbf{E}^{n+1} = \left[1 - \alpha(1-\alpha)(c\Delta t)^2 \nabla \times \nabla \times \right] \mathbf{E}^n + c\Delta t \nabla \times \mathbf{B}^n - 4\pi\Delta t \mathbf{j}^{n+1/2}. \quad (12)$$

The current density in the equation (12) is unknown and needs to be predicted, the prediction is done with aid of the equations of motion:

$$\mathbf{j}^{n+1/2}(\mathbf{x}) = \sum_j e_j \mathbf{v}_j^{n+1/2} S(\mathbf{x} - \mathbf{x}_j^{n+1/2}) \quad (13)$$

$$= \sum_j e_j \left[\mathbf{v}_j^n + \gamma \Delta t \frac{e_j}{m_j} [\mathbf{E}^{n+0} + \left\{ \frac{\mathbf{v}_j^n}{c} + \frac{\Delta t}{2} \frac{e_j}{m_j c} \left(\mathbf{E}^{n+0} + \frac{\mathbf{v}_j^n}{c} \times \mathbf{B}^{n+0} \right) + \frac{\Delta t}{2} \frac{e_j}{m_j c} \mathbf{B}^{n+0} \times \mathbf{B}^{n+0} \right\} \times \frac{\mathbf{B}^{n+0}}{1 + \left(\frac{\Delta t e_j E}{2 m_j c} \right)^2}] S(\mathbf{x} - \mathbf{x}_j^{n+1/2}) \right. \\ \left. + \sum_j (-e) \left[(v_\parallel^n + \gamma \Delta t \left(\frac{-e}{m_e} \mathbf{E}_\parallel^{n+0}(\mathbf{x}_j^{n+1/2}) - \frac{\mu_B}{m_e} \nabla_\parallel \mathbf{B}^{n+0} \right)) \mathbf{b}^{n+1/2} + v_\parallel^{n+1/2}(\mathbf{x}_j^{n+1/2}) \right] S(\mathbf{x} - \mathbf{x}_j^{n+1/2}) \right] \quad (14)$$

Substitution of this expression into Eq.(12) with (11) and the equations of motion forms a closed set of the Courant-condition-free, implicit equations. This equation is named "directly-coupled field-particle equation." The detail of the solution is found in Ref.7.

3. Applications

As the first physics application of the macroscale particle code, the Alfvén-ion-cyclotron (AIC) instability is shown which is driven unstable by the ion temperature anisotropy. The magnetic field energy grows exponentially from the initial low noise level and saturates around $\omega_{ci}t \sim 40$. The ion temperature anisotropy, $(T_{\perp}/T_{\parallel})_i$, decreases in the nonlinear stage. The final value of the anisotropy, $(T_{\perp}/T_{\parallel})_i \approx 2$ is found to be almost independent of the initial ones, as found before.

Ion scatter plots in the (v_x, v_z) and (z, v_z) spaces are shown in Fig.1. The ions suffer from a remarkable pitch angle scattering during $\omega_{ci}t \sim 30 - 40$. This is most clearly seen as the change in the (v_x, v_z) ρ_{\perp} - ρ_{\parallel} distribution from the needle-like distribution into a more isotropic round-shaped one at later times. This phenomenon has been observed as the decrease in the temperature anisotropy. The overall results of the AIC instability of the present simulation have agreed well with the past hybrid simulation⁸⁾. Moreover, the macroscale particle simulation has been found to be numerically more stable⁷⁾.

As the second example, the full-implicit macroscale particle code is applied to the kink instability of a current-carrying ion beam which has an inhomogeneous density profile. The simulation is performed in the three-dimensional space. The initial drift speed is given to the ions so that the safety factor at the center becomes $q(0) \approx 0.5$. In the simulation a helical distortion of the ion beam becomes apparent from $\omega_{ci}t \sim 40$. The displacement due to this distortion takes a maximum value around $\omega_{ci}t \sim 130$ and stays at this level thereafter.

Fig.2 is the bird's-eye view plot of the ion distribution in the (x, y, z) space. At $\omega_{ci}t \sim 140$, the $m/n=1/1$ helical distortion is clearly observed. Spatially the same helical distortion is observed for the electrons. This fact implies an occurrence of the helical distortion of the magnetic field structure, as it is actually proved. Since the above helical distortion occurs in a few poloidal Alfvén times, it is considered to be an ideal magnetohydrodynamic phenomenon - the ideal kink instability.

References

1. J.U. Brackbill and D.W. Forslund, J.Comput.Phys., 46, 271 (1982).
2. H.X. Vu and J.U. Brackbill, Comput.Phys.Comm., submitted (1991).
3. M. Tanaka, J.Comput.Phys., 79, 209 (1988).
4. M. Tanaka, T. Sato and A. Hasegawa, Phys. Fluids, B1, 325 (1989).
5. J.F. Federici, W.W. Lee and W.M. Tang, Phys. Fluids, 30, 425 (1987).
6. M. Tanaka, J. Geophys. Res., 90, 6459 (1985).
7. M. Tanaka, S. Murakami, H. Takamaru and T. Sato, NIFS Report No.91 (National Institute for Fusion Science) (1991).

Table 1 Characteristics of Macroscale Particle Simulation Code

<ul style="list-style-type: none"> • Large space and time scales: $\omega_{pi}\Delta t \gg 1$, $\omega_{ci}\Delta t \gg 1$, and $\Delta x/(c/\omega_{pi}) \geq 1$ • Fully electromagnetic. • Multi-dimensions in any geometry (Cartesian, cylinder, torus). • Inhomogeneous density profile and arbitrary magnetic field structure. • Fully kinetic: <ul style="list-style-type: none"> ions: <ul style="list-style-type: none"> electrons: <ul style="list-style-type: none"> 3-D motion parallel direction \rightarrow 1-D motion with $(-\mu \nabla_{\parallel} B)$ force perpendicular direction \rightarrow guiding-center approximation ($\mathbf{E} \times \mathbf{B}$, ∇B, curvature drifts) Resonance effects (Landau, cyclotron resonances) Orbit effects finite Larmor radius effects diamagnetic current banana orbit... Longitudinal particle transport 	
included	

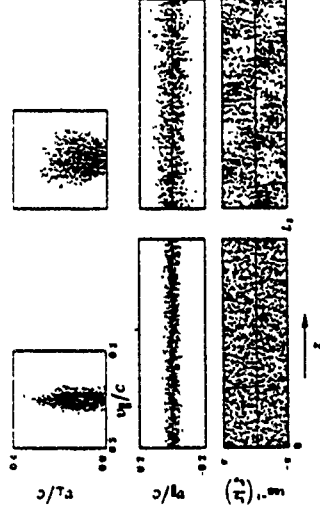


Figure 1

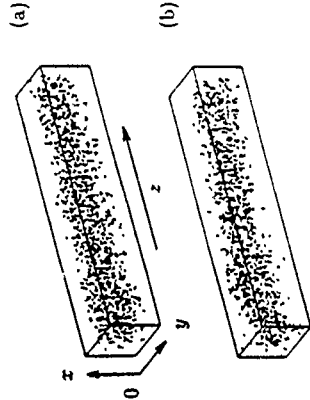


Figure 2

AN INTERPARTICLE COLLISION MODEL FOR ELECTROMAGNETIC HYBRID SIMULATIONS OF HIGH DENSITY PLASMAS*

Michael E. Jones, V. A. Thomas, and D. Winske
MS-F645, Los Alamos National Laboratory,
Los Alamos, NM 87545

Particle-in-cell methods have long been used to model collisionless plasma phenomena. At the other extreme, fluid methods have been used to model highly collisional plasmas. Hybrid models in which ions are treated by the particle-in-cell method and electrons are treated as a massless fluid allow one to ignore phenomena on the electron plasma frequency time scale and thus have proven to be useful for modeling certain aspects high density plasmas. Until now particle collisions in PIC or hybrid models have been limited to scattering of the particles off a fixed background.

The key ingredient in developing plasma simulation methods that allow one to study the transitional region between collisionless and collisional plasmas has been the development of an adequate interparticle collision model that takes into account the self-consistent reaction of the colliding species. We have developed and incorporated one type of interparticle collision model into the hybrid version of our particle-in-cell code ISIS.¹ The method is highly efficient and ensures local conservation of momentum and energy.

Monte Carlo Scattering Force

The main problem with a self-consistent scattering model in a particle-in-cell code is the question of how to conserve momentum and energy locally. In the case of scattering from a fixed background, the reaction of the background is not needed, so the usual random number based Monte Carlo scattering may be used. However, if we need the self-consistent scattering of two or more species, random scattering treating the scattering species as a fixed background will not conserve momentum and energy unless very large numbers of particles are used.

Another approach that has been suggested is to sort the particles and perform scattering in pairs to conserve momentum and energy. The sorting process can be time consuming and a prescription for which particles to pair together is not obvious. Molecular dynamics modeling uses this approach to describe the interactions. In general, particles are seldom exactly in the same location at a given time step, so a decision of when to scatter is not clear. Lattice gas algorithms solve this problem by requiring all the particles move at the same velocity.

The approach to collision modeling presented here is a natural extension of the particle-in-cell method. It involves defining a mesh quantity for the collision operator that can be viewed as a "collision field". This field will mediate the collision force and we will require that this force give local conservation of momentum and energy. Define $F_{\alpha\beta}$ as the force on a particle in species α due to collision with the particles of species β . Momentum conservation requires

* This work was performed under the auspices of the U.S. Dept. of Energy

$$(\mathbf{F}_{\alpha\beta}) = -(\mathbf{F}_{\beta\alpha}), \quad (1)$$

where the brackets denote distribution function averages. For example, the density, n_α and velocity, $\langle \mathbf{v}_\alpha \rangle$ moments are given by

$$n_\alpha \equiv \int f_\alpha(\mathbf{v}) d^3v \quad \text{and} \quad \langle \mathbf{v}_\alpha \rangle \equiv \frac{1}{n_\alpha} \int f_\alpha(\mathbf{v}) \mathbf{v} d^3v. \quad (2)$$

We will also require that our collision force conserve energy locally:

$$(\mathbf{v}_\alpha \cdot \mathbf{F}_{\alpha\beta}) = -(\mathbf{v}_\beta \cdot \mathbf{F}_{\beta\alpha}), \quad (3)$$

Equations (1) and (3) represent conservation laws only in an average sense. If we perform the integral in Eq. (2) by weighted sums over the particles, these restrictions ensure that the first three moments of the distribution function (density, velocity, and temperature) are conserved. Note that we are not requiring microscopic (every particle) conservation laws. We will furthermore assume that the force is a function only of the first 3 moments and the particle velocity, i.e.,

$$\mathbf{F}_{\alpha\beta} = \mathbf{F}_{\alpha\beta}(n_\alpha, n_\beta, \langle \mathbf{v}_\alpha \rangle, \langle \mathbf{v}_\beta \rangle, \langle \mathbf{v}_\alpha^2 \rangle, \langle \mathbf{v}_\beta^2 \rangle, \mathbf{v}_\alpha). \quad (4)$$

There are many possible choices of $\mathbf{F}_{\alpha\beta}$ which satisfy Eqs. (1) and (3). We have considered the following simple form of the collision force

$$\mathbf{F}_{\alpha\beta} = \nu_1 n_\beta (\langle \mathbf{v}_\beta \rangle - \langle \mathbf{v}_\alpha \rangle) + \nu_2 n_\beta \Delta T_{\beta\alpha} \left\{ \frac{\mathbf{v}_\alpha - \langle \mathbf{v}_\alpha \rangle}{|\langle \mathbf{v}_\alpha \rangle|} \right. \\ \left. - \nu_3 n_\beta (\langle \mathbf{v}_\beta \rangle - \langle \mathbf{v}_\alpha \rangle) \times \mathbf{R} + \nu_4 n_\beta (\langle \mathbf{v}_\beta \rangle - \langle \mathbf{v}_\alpha \rangle), \right. \quad (5)$$

where the temperature difference $\Delta T_{\beta\alpha} \equiv m_\beta (\langle \mathbf{v}_\beta^2 \rangle - \langle \mathbf{v}_\alpha \rangle^2) - m_\alpha (\langle \mathbf{v}_\alpha^2 \rangle - \langle \mathbf{v}_\beta \rangle^2)$. The parameters ν_1, ν_2, ν_3 , and ν_4 may depend only on the local moments. The vector \mathbf{R} is generated from random numbers. We have not developed a first principles argument for this form of the force, rather it was chosen from somewhat heuristic arguments. The first term provides a drag due to the relative motion of the two species and corresponds to the Krook model used for fluid equations. The second term provides for temperature equilibration between the two species. The third term represents a scattering operator that on the average does not change the local momentum or energy. The last term is similar to the first and is added to ensure conservation of energy.

It is straightforward to show that momentum conservation as defined by Eq. (1) is satisfied by the force in Eq. (5) for an arbitrary choice of the parameters. The energy conservation equation, Eq. (3), places a constraint on the parameters which we write as

$$\nu_4 = \frac{-1}{(\langle \mathbf{v}_\alpha^2 \rangle + \langle \mathbf{v}_\beta^2 \rangle - 2\langle \mathbf{v}_\alpha \rangle \cdot \langle \mathbf{v}_\beta \rangle)} \left[\nu_1 (\langle \mathbf{v}_\beta \rangle - \langle \mathbf{v}_\alpha \rangle)^2 \right. \\ \left. + \nu_2 \Delta T_{\beta\alpha} \left(\frac{\langle \mathbf{v}_\alpha^2 \rangle - \langle \mathbf{v}_\beta \rangle^2}{|\langle \mathbf{v}_\beta \rangle|} - \frac{\langle \mathbf{v}_\alpha^2 \rangle - \langle \mathbf{v}_\alpha \rangle^2}{|\langle \mathbf{v}_\alpha \rangle|} \right) \right] \quad (6)$$

The other parameters, ν_1 , ν_2 , and ν_3 are unconstrained. The first parameter is related to the rate at which momentum is exchanged, the second parameter is related to the rate at which the temperatures equilibrate and the last parameter is related to the rate of angular scattering. The scattering term is complicated by the fact that this term can also lead to momentum and energy exchange if the mean free path of the particles are large enough. A prescription for how to choose these parameters to model a given physical situation is still under investigation. In general the model, however, exhibits the correct behavior for the processes of momentum and energy exchange and scattering. It contains enough free parameters that most physical situations can be modeled.

It should be noted that there is nothing in the formalism that prevents the collision force from being applied to electron-ion collisions and/or self-collisions. In the hybrid model, the electrons are treated as a massless fluid. Therefore, the force on the electrons is the grid averaged force. Furthermore, the scattering term time averages to zero. In this limit only the first and last terms in Eq. (5) survive and the force on the electron fluid is

$$(\mathbf{F}_e) = \sum_{\beta} \mathbf{F}_{\alpha\beta} = \sum_{\beta} (\nu_1 + \nu_2) n_{\alpha} n_{\beta} ((\mathbf{v}_{\beta}) - (\mathbf{v}_{\alpha})), \quad (7)$$

which is of the same form as the usual Krook term.

Collisions for like particles within a species in this formalism, do not exchange momentum and energy directly. For $\beta = \alpha$, the only term that survives is the third term in Eq. (5), or the scattering term

$$\mathbf{F}_{\alpha\alpha} = \nu_3 n_{\alpha} ((\mathbf{v}_{\alpha}) - (\mathbf{v}_{\alpha})) \times \mathbf{R}. \quad (8)$$

The effect of this term is to provide a rotation of the distribution. This does not mean, however, that momentum and energy cannot be exchanged within a species. The exchange occurs through spatial mixing caused by the rotation.

Numerical Implementation

For computational convenience it is desirable to regroup the terms in Eq. (5) as follows

$$\mathbf{F}_{\alpha\beta} = \mathbf{F}_{\alpha\beta}^{(1)} + \mathbf{F}_{\alpha\beta}^{(2)} \mathbf{v}_{\alpha} + \mathbf{v}_{\alpha} \times \mathbf{N}, \quad (9)$$

where

$$\mathbf{F}_{\alpha\beta}^{(1)} \equiv (\nu_1 + \nu_2) n_{\alpha} (\mathbf{v}_{\beta}) - (\nu_1 + \nu_2 \frac{\Delta T_{\beta\alpha}}{[(\mathbf{v}_{\alpha})]}) n_{\beta} (\mathbf{v}_{\alpha}) - \nu_3 n_{\beta} (\mathbf{v}_{\beta}) \times \mathbf{R}, \quad (10)$$

$$\mathbf{F}_{\alpha\beta}^{(2)} \equiv (\nu_2 \frac{\Delta T_{\beta\alpha}}{[(\mathbf{v}_{\alpha})]} - \nu_2) n_{\beta}, \quad (11)$$

and

$$\mathbf{N} \equiv \nu_3 n_{\beta} \mathbf{R}. \quad (12)$$

In this form we can see that the first term is equivalent to an electric field and the last term is equivalent to a magnetic field. Thus we can use the usual Boris particle pusher.

We treat the term proportional to \mathbf{v}_{α} as an additional electric field term after the first half and before last half of the electric field update. These forces are in fact added as a separate gather operation on the particles. The velocity update takes place with both real fields and collision fields simultaneously.

Tests of the Method

As an example of the test of the collision model, Fig. 1 shows the distribution function of two species interacting via our model. The particles are initialized as overlapping spheroids in 2-d cylindrical (r-z) coordinates. The parameters are chosen so that equilibration occurs before much separation.

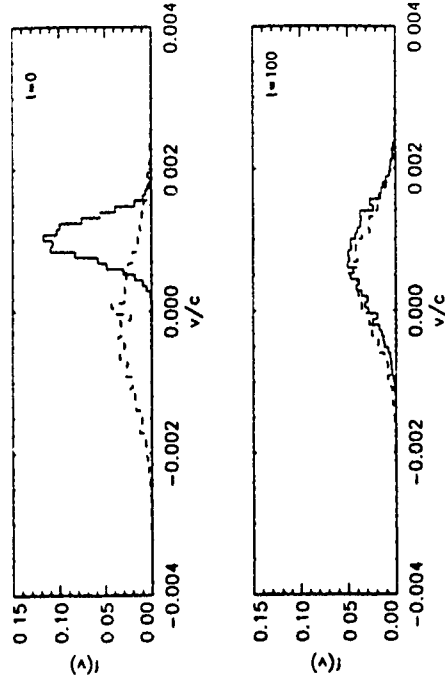


Fig. 1 Evolution of distribution functions for two colliding species in the absence of fields. The colder drifting Maxwellian equilibrates with the warmer stationary Maxwellian by exchanging momentum and energy.

Acknowledgments

The authors wish to acknowledge useful discussions with R. D. Jones and I. R. Luomala.

References

1. M. E. Jones, V. A. Thomas, R. J. Mason and D. Winske, "A Fully Electromagnetic Hybrid Model for High Density Plasma Simulations," Proceedings of the 13th Conf on the Num. Sim. of Plasmas, PMA15, Santa Fe, NM, Sep 17-20, 1989

Gyrokinetic Particle Simulation of Ion Temperature Gradient Instabilities in 3D Toroidal Geometry

S E Parker and W. W. Lee

Plasma Physics Laboratory, Princeton University, Princeton NJ 08543

1 Overview

We have developed a three dimensional electrostatic toroidal gyrokinetic particle code to study tokamak transport. The whole tokamak cross section is modeled inside a square cross section toroidal grid. Here we present the computational model and η mode simulation results. We will show both the linear and fully nonlinear evolution of potential, density and heat flux, and study the effects of trapped and passing ion orbits on linear stability and the saturation of these quantities. A fully linearized version of the code has been developed and is currently being benchmarked against 2D eigenmode calculations[1]. In the future, we plan to add drift kinetic and bounced averaged electrons, and finite beta effects (δB) to the toroidal code greatly extending it's applicability.

2 Model Equations

We assume the system is electrostatic and the electrons are adiabatic ($\delta n_e = n_0 e \phi / T_e$). The ions are evolved with the full toroidal electrostatic gyrokinetic equations of motion [2]

$$\begin{aligned} \mathbf{x} = v_{\parallel} \hat{\mathbf{b}}_0 + \frac{v_{\perp}^2}{\Omega_0} \hat{\mathbf{b}}_0 \times (\hat{\mathbf{b}}_0 \cdot \nabla) \hat{\mathbf{f}}_0 + \frac{v_{\perp}^2}{2\Omega_0} \hat{\mathbf{b}}_0 \times \nabla \ln B_0 + \frac{c}{B_0} \hat{\mathbf{E}}_0 \times \nabla \phi - \frac{q}{m\Omega_0} \hat{\mathbf{b}}_0 \times \mathbf{K} \phi, \quad (1) \\ v_{\parallel} = - \left[\hat{\mathbf{b}}_0 + \frac{v_{\parallel}}{\Omega_0} \hat{\mathbf{b}}_0 \times (\hat{\mathbf{b}}_0 \cdot \nabla) \right] \cdot \left[\frac{v_{\perp}^2}{2} \nabla \ln B_0 + \frac{q}{m} \nabla \phi \right]. \quad (2) \end{aligned}$$

where the last term in Eq. (1) is a multiple scale expansion used to model the equilibrium spatial temperature and density profiles. B_0 and $\hat{\mathbf{b}}_0$ are the fixed equilibrium magnetic field magnitude and direction. We use a slab like coordinate system which is suitable for Fourier decomposition of the field equation. The coordinates (x, y, z) in terms of the usual toroidal coordinates (r, θ, ψ) are: ($x = r \cos \theta, y = r \sin \theta, z = -(R_0 - a)\psi$). Using these coordinates and assuming $k_{\parallel} \ll k_{\perp}$ one can transform the field equation[3] to obtain:

$$-\frac{\gamma}{\chi B_0} (1 - \Gamma_0(k_{\perp} \rho_s)) \phi(k_{\perp}, z) = -4\pi e \left[\delta \tilde{n}_i(k_{\perp}, z) - n_0 \frac{e \phi(k_{\perp}, z)}{T_e} \right], \quad (3)$$

where higher order terms have been neglected. For the radial boundary condition we set the right hand side of Eq. (3) to zero for $r \geq a$.

3 Simulation Results

First, we show a linearized cylindrical η mode simulation with parameters chosen to compare with the Columbia Linear Machine[2]. In this case, the simulation region is straight (i.e. no toroidal metric). Figures 1 (A) and (B) are contour plots of two slices of the 3D potential, $\phi(x, y, z = 0)$ and $\phi(x, y = L_y/2, z)$. The $m = 2, n = 1(kz = 2\pi/L_z)$ mode is most unstable. We also have modeled poloidal rotation by adding an equilibrium radial electric field.

Secondly, we show a toroidal run with $q = 1.1 - 1.9$, keeping only the $n = 2$ toroidal mode, hence there is only one rational surface at $q = 1.5$. Figure 2 (A) and (B) are four guiding center particle trajectories in an equilibrium toroidal field (different parameters than Figure 3). Figure 3 (A) and (B) are contour plots of $\text{Re } \phi(x, y, n = 2)$ and $|\phi|(x, y, n = 2)$. Note the formation of the $m = 3$ ($q = 1.5 = m/n = 3/2$) mode with significant ballooning on the outside of the torus.

Acknowledgments

This work is supported by the U.S. DOE Fusion Postdoc Program and DOE Contract No. DE-AC02-76-CHO-3073.

References

- [1] R. Marzhan, W.M. Tang and G. Rewoldt, *Phys. Fluids* **23** 1164 (1980).
- [2] T.S. Hahn, *Phys. Fluids* **31** 2670 (1988).
- [3] W. W. Lee, *J. Comput. Phys.* **72**, 243 (1987).
- [4] A K. Sen, J. Chen and M. Mauel *Phys. Rev. Let.* **66** 429 (1991).

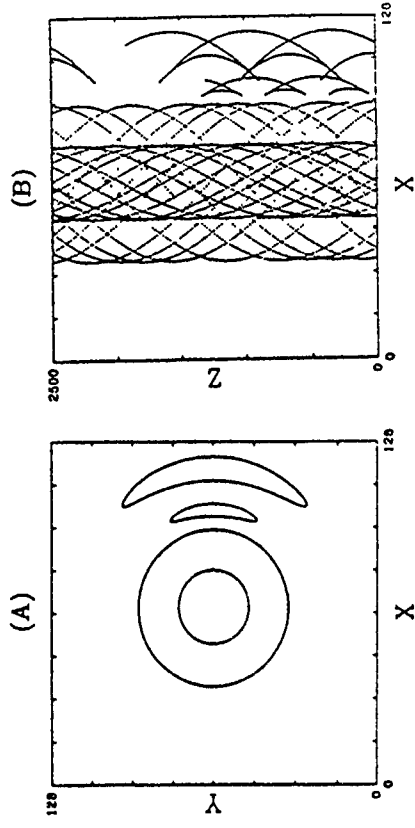


Figure 2 (A) and (B): Particle trajectories; (A) $\phi(x, y)$, and (B) $\phi(x, z)$

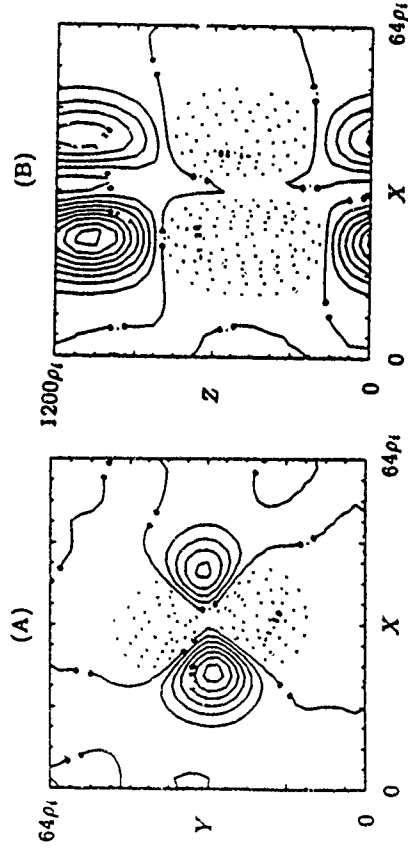


Figure 3 (A) and (B): Cylindrical η mode potential contours, (A) $\phi(x, y, z = 0)$, (B) $\phi(x, y = L_y/2, z)$.

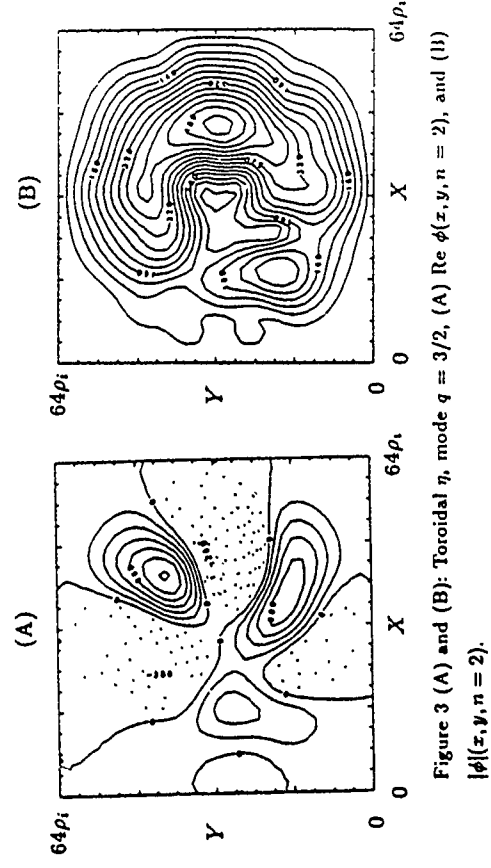


Figure 4 (A) and (B): Toroidal η mode $q = 3/2$, (A) $\text{Re } \phi(x, y, n = 2)$, and (B) $|\phi|(x, y, n = 2)$.

New Developments in FLIP-MHD: A Particle-In-Cell Method for Magnetohydrodynamics

J U Brackbill
Los Alamos National Laboratory
Los Alamos, NM 87545

Introduction

There have been new developments and applications of FLIP-MHD, a particle-in-cell method for magnetohydrodynamics. These include a mass-matrix formulation to reduce or eliminate the computational diffusion caused by double interpolation, applications to space and solar physics problems, and comparison with more conventional methods for MHD flow.

Review of the Method

FLIP-MHD extends FLIP [1] to magnetohydrodynamic (MHD) flow in two dimensions[2]. Particle motion replaces finite difference approximations to convective transport, which reduces computational diffusion of the magnetic field.

FLIP is an extension of "classical" PIC [3], where particles have mass, but every other property of the fluid is stored on a grid. In FLIP, particles have every property of the fluid, so that they provide a complete Lagrangian description not only to resolve contact discontinuities, but also to reduce computational diffusion of linear and angular momentum[4, 5]. The interactions among the particles are calculated on a grid, for convenience and economy. The present study extends FLIP to MHD, by including information about the magnetic field among the attributes of the particles.

FLIP-MHD approximates the equations for viscous/resistive MHD flow comprising a mass continuity equation, Faraday's law, a momentum equation, and an energy equation. The algorithm for solving the MHD equations using the particle-in-cell method is similar to other PIC algorithms:

- 1 Represent the fluid and the field by assigning appropriate properties to finite-sized particles. Interpolate the particle data on to a grid to initialize the dependent variables τ , u , B , and p at the grid points.
- 2 Solve finite-difference approximations to the MHD equations on the grid to advance the solution from t to $t+\Delta t$
- 3 Interpolate the solutions of MHD equations on the grid back to the particles.
- 4 Move the particles through the grid to model convection.

One can illustrate these four steps by tracing the calculation of the magnetic field through a computation cycle.

Step 1. In FLIP-MHD, a magnetic field is represented by assigning each particle p a

magnetic moment, μ_p . From μ_p , one calculates a magnetization by interpolation from the particles to a grid of quadrilateral cells, each one labeled by an index c with volume V_c ,

$$M_c V_c = \sum_p \mu_p S_{pc} \quad (1)$$

where S_{pc} is the interpolation weight, which is a function of the distance between the particle at x_p and the cell center at x_c . S is typically a B-spline, which has several useful properties[6]. Among them, when $|x_c - x_p| > O(\Delta x)$, S is zero. This reduces the computational cost of interpolation. When S is summed over all the cells c on the grid, the sum is equal to one. This aids in formulating conservative methods.

A solenoidal magnetic field is calculated from the magnetization by subtracting the gradient of a scalar potential,

$$B = M - \nabla \phi \quad (2)$$

where ϕ is given by,

$$\nabla \cdot M = \nabla^2 \phi \quad (3)$$

Laplace's equation for ϕ is solved using a conjugate gradient iteration[7].

Step 2. The evolution of the magnetic field is calculated by solving finite difference approximations to Faraday's law,

$$\frac{d}{dt} \left[\frac{B}{\rho} \right] = \frac{B}{\rho} \cdot \nabla u + \frac{1}{\rho} [\nabla \times \eta J] \quad (4)$$

$$\frac{4\pi J}{c} = \nabla \times B \quad (5)$$

where B is the magnetic field intensity, J is the current density, c is the speed of light, u is the fluid velocity, and η is the resistivity. The solenoidal condition on B , $\nabla \cdot B = 0$, is imposed as an auxiliary condition on the solution.

To advance the magnetization from t to $t+\Delta t$, one solves an evolution equation for M ,

$$\frac{dM}{dt} = \frac{dB}{dt} - (\nabla u) \cdot \nabla \phi \quad (6)$$

Step 3. At this point in a computation step, one has advanced the magnetization on the grid, but not the particle magnetic moments. Differentiating Eq. (1) yields an implicit evolution equation for the particle magnetic moment,

$$\frac{dM_c V_c}{dt} = \sum_p \frac{d\mu_p}{dt} S_{pc} \quad (7)$$

Because there are more particles than grid points, one cannot simply solve for the particle magnetic moment. Earlier, an approximate inverse to Eq. (7) was introduced, which resulted in a some diffusion. Using a mass-matrix[8, 9], this small diffusion can be reduced still further.

In the mass matrix formulation, a sharpened magnetization is calculated from the implicit equation,

References

1. Brackbill, J.U. & Ruppel, H.M. *J. Comput. Phys.* **65**, 314 (1986).
2. Brackbill, J.U. *J. Comput. Phys.* (1991-to appear).
3. Harlow, F.H. In *Fundamental Methods in Hydrodynamics* (eds Alder, B., Fernbach, S. & Rotenberg, M.) 319-345 (Academic Press, New York, 1964).
4. Brackbill, J.U. *Comput. Phys. Comm.* **47**, 1-16 (1987).
5. Brackbill, J.U., Kothe, D.B. & Ruppel, H.M. *Comput. Phys. Comm.* **48**, 25-38 (1988).
6. Boor, C.D. *A Practical Guide to Splines* (Springer-Verlag, 1978).
7. Jordan, T.J. In *Parallel Computations* (eds. Rodrigue, G.) **1** (Academic Press, Orlando, 1982).
8. Burgess, D., Sulsky, D. & Brackbill, J.U. *Journal of Computational Physics* (1991-submitted).
9. O'Rourke, P.J., Larratrou, B. & Brackbill, J.U. *Journal of Computational Physics* (1991-submitted).
10. Meltz, B.J.A. *Journal of Computational Physics* (1990-submitted).
11. Colella, P. & Woodward, P.R. *Journal of Computational Physics* **54**, 174-201 (1984).
12. vanLeer, B. *Journal of Computational Physics* **32**, 101 (1979).

$$\sum_c \left\{ \frac{dM_c V_c}{dt} \right\} T_{cc'} = \frac{dM_c V_c}{dt} \quad (8)$$

The mass matrix, $T_{cc'}$, is defined by,

$$T_{cc'} = \sum_p S(x_p - x_c) S(x_p - x_{c'}) \quad (9)$$

The change in the particle magnetic moment is then calculated by interpolating the sharpened magnetization.

$$\frac{d\mu_p}{dt} = \sum_c \left\{ \frac{dM_c V_c}{dt} \right\} S_{pc} \quad (10)$$

Using Eqs. (8) and (10), one can easily show that Eq. (7) is satisfied.

One advantage of introducing the mass matrix is that the grid and particle magnetic energy changes are equal. One can show this using Eq. (10). Recall the definitions of the two magnetic energies,

$$\sum_p \theta_p = \sum_p \frac{1}{2} \mu_p \cdot \sum_c B_c S_{pc} = \sum_c \frac{1}{2} B_c \cdot M_c V_c = \sum_c E_c \quad (11)$$

With the mass matrix formulation, Eq. (8), the difference in the rate of change of the particle and grid energies is zero,

$$\sum_p \frac{d\theta_p}{dt} - \sum_c \frac{dE_c}{dt} = \frac{1}{2} \sum_c B_c \cdot \left[\sum_c \frac{d(M_c V_c)}{dt} T_{cc'} - \frac{dM_c V_c}{dt} \right] = 0 \quad (12)$$

Although one must invert the mass matrix to solve Eq. (8), and although the mass matrix is a 5×5 matrix at each grid point (with quadratic interpolation in 2 dimensions), it is not too expensive to do so. The mass matrix is symmetric and diagonally dominant, and a simple conjugate gradient iteration converges rapidly. However, the major cost of the mass matrix formulation is in forming and storing the matrix, which requires summing particle contributions. For that reason, one can partially lump the mass matrix by adding the contributions for which c and c' differ by 2 to the diagonal of the matrix. A simple analysis shows the partially lumped mass matrix has a numerical diffusivity that is 25% of the diffusivity with the approximate inverse, which was estimated by imposing a physical resistivity that doubles the diffusion[2]. By computation, the diffusivity needed to double the diffusion corresponds to a grid magnetic Reynolds number equal to 17. With the partially lumped formulation, the grid magnetic Reynolds number is estimated in the same way to be 64. (By comparison, for an ordinary Eulerian calculation the value of the grid Reynolds number is approximately 1.)

Results

The maximum Ohmic heating in a confined eddy calculated with FLIP-MHD [2] has been compared with standard methods for modeling convection, such as Van Leer and a third-order method similar to PPM on a confined eddy[10, 11, 12]. The results are summarized in Fig. 1, where heating is plotted against spatial resolution. The lower dissipation with FLIP results in higher maximum currents, and greater Ohmic heating.

The results of comparisons of results with FLIP with standard finite difference methods will be shown for reconnection in the earth's magnetotail, and with reconnection occurring due to a Kelvin-Helmholtz instability.

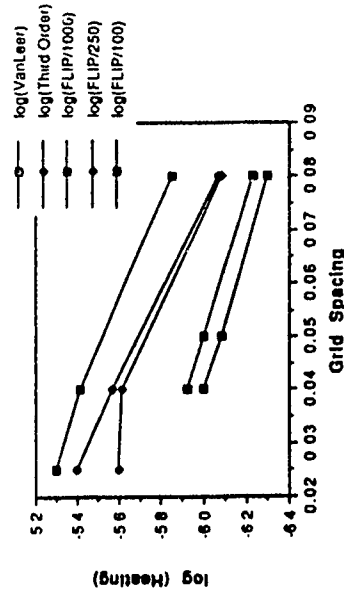


Fig. 1. Results of a confined eddy calculation with a van Leer and a third order finite difference approximation of convection are compared with FLIP-MHD. The Ohmic heating decreases with increasing dissipation and is a sensitive measure of error. The prescribed magnetic Reynolds number, R_m , for both finite difference methods is 1000. The agreement between FLIP and van Leer is improved when the prescribed R_m in FLIP is decreased to 100, and between FLIP and third order when the prescribed R_m is decreased to 250, at least for grid spacings > 0.04 . (The FLIP calculation with $R_m=250$ has converged with grid spacing < 0.04).

Address List of Preregistered Attendees

Jean Claude Adam
Ecole Polytechnique
Centre de Physique Theoretique
91128 Palaiseau Cedex
FRANCE

Alain Adolf
CEA-CELV
B.P. 27
91195 Villeneuve-St. Georges-Cedex
FRANCE

John Ambrosiano
LLNL
P.O. Box 808, L-95
Livermore, CA 94550

David Anderson
LLNL
MS L-561
PO Box 808
Livermore, CA 94550

Wayne Arter
AEA Technology
Culham Laboratory
Abingdon, Oxfordshire OX 14 3DB
ENGLAND

Dan Barnes
LANL
P O Box 1663
MS F647, T-15
Los Alamos NM 87545

Glenn Bateman
PPPL
P O. Box 451
Princeton, NJ 08543

Igor Bezmenov
USSR Academy of Sciences
Keldysh Inst. of Appl. Mathematics
Musskaya Sq. 4
Moscow A-47
125047 USSR

Charles K. Birdsall
Univ. of California
Cory Hall
EECS Dept.
Berkeley, CA 94720

Arthur Bland
Martin Marietta
P.O. Box 2003
Oak Ridge, TN 37831-7001

Serge Bouquet
CEA
Centre de Vaujours-Moronvilliers (CEV-M)
BP 7
77181 Courtry FRANCE

Jerry Brackbill
LANL
P.O. Box 1663, Group T-3, B-216
Los Alamos, NM 87545

Scott Brandon
LLNL
P.O. Box 808, MS L-95
Livermore, CA 94550

Debra Callahan
LLNL
7000 East Avenue, MS L-794
Livermore, CA 94550

Edward Caramana
LANL
X-1, MS-F645
Los Alamos, NM 87545

Peter Cargill
Univ. of Maryland
Department of Physics and Astronomy
College Park, MD 20742

Bruce Church
Cornell Univ.
Laboratory of Plasma Studies
369 Upson Hall
Ithaca, NY 14853

Florence Daverdon
CEA/CESTA
Service DT/PE/PHEP
Route des Garguils
BP 2
33114 Le Barp, FRANCE

Pierre Degond
CEA/CELV
DMA/MCN
BP 27
94195 Villeneuve-St. Georges-Cedex
FRANCE

Jacques Denavit
LLNL
P.O. Box 808, L-95
Livermore, CA 94550

Andris Dimits
LLNL
P.O. Box 5511
Livermore, CA 94550

<p>Bill Dorland PPPL Princeton University P.O. Box 45 Princeton, NJ 08543</p> <p>Adam Drobot SAIC 1710 Goodridge Drive McLean, VA 22102</p> <p>Ken Eppley SLAC Bin 26 P.O. Box 4349 Stanford, CA 94309</p> <p>Rickey Faehl LANL P.O. Box 1663, X-10, MS-B259 Los Alamos, NM 87545</p> <p>Marc Feix CNRS 3A Avenue de la Recherche Scientifique 45071 Orleans Cedex 2 FRANCE</p> <p>Bob Ferraro JPL 4800 Oak Grove Drive Pasadena, CA 91109-8099</p> <p>Eric Fijaikow CNRS PMMS 3A Avenue de la Recherche Scientifique 45071 Orleans Cedex 2 FRANCE</p> <p>Dave Forslund LANL MS B287 Los Alamos, NM 87545</p>	<p>Henry Freund SAIC 1710 Goodridge Drive McLean, VA 22102</p> <p>Alex Friedman LLNL L-472 PO Box 5508 Livermore, CA 94550</p> <p>Geoffry Furnish Univ. of Texas Institute for Fusion Studies Austin, TX 78712</p> <p>Keshab Ganguly Texas Tech Univ. Dept. of Mathematics Lubbock, TX 79409</p> <p>Alain Ghizzo CCFM 1804 Montee Ste. Julie Varenes J3X151 Quebec CANADA</p> <p>Ron Ghosh NASA Goddard Code 692 USRA Greenbelt, MD 20771</p> <p>Alan Glasser LANL P.O. Box 1663 T-15, MS F-647 Los Alamos, NM 87545</p> <p>Chuck Goodrich Univ of Maryland Astronomy Department College Park, MD 20742</p>	<p>Mark Gray Univ. of Texas Institute for Fusion Studies Austin, TX 7871</p> <p>David Grote LLNL P.O. Box 5508, L-637 Livermore, CA 94550</p> <p>Irv Haber NRL Washington, DC 20375-5000</p> <p>Greg Hammett PPPL Princeton University P.O. Box 451 Princeton, NJ 08543</p> <p>James Harold Univ. of Maryland Laboratory for Plasma Research College Park, MD 20742</p> <p>Enrique Henestroza LBL 1 Cyclotron Road, MD 47-112 Berkeley, CA 94720</p> <p>Anne Heron Ecole Polytechnique Centre de Physique Theoretique 91128 Palaiseau Cedex FRANCE</p> <p>Dennis Hewett LLNL 7000 East Avenue Livermore, CA 94550</p>
---	---	---

Robert Holdaway
Univ. of Iowa
Department of Physics and Astronomy
Iowa City, IA 52242

Barry Howard
NERSC
P.O. Box 5509
Livermore, CA 94550

Richard Hubbard
NRL
Code 4792
Washington, DC 20375-5000

John Johnson
PPPL
Princeton University
PO Box 451
Princeton, NJ 08543

Michael Jones
LANL
MS-F645
Los Alamos, NM 87545

Glenn Joyce
NRL
Code 4792
Washington, DC 20375-5000

Zerefsan Kaymaz
UCLA
Department of Atmospheric Science
Los Angeles, CA 90024

Mark Kieler
SNL
Division 1265
P.O. Box 5800
Albuquerque, NM 87185

Chris Kostas
SAIC
1710 Goodridge Drive
McLean, VA 22102

Michael Kotschenreuther
Univ. of Texas
Institute for Fusion Studies
RLM 11.222
Austin, TX 78712

Jon Krall
NRL
Plasma Physics Division
Beam Physics Branch
Washington, DC 20375-5000

Mike Kress
CUNY
116 City Boulevard
Staten Island, NY 10301

Warren Krueger
SAIC
1710 Goodridge Drive
McLean, VA 22102

Serge Kuzmin
USSR Institute of Nuclear Physics
ul. Lavrentjeva, 11
Novosibirsk, 630090
USSR

Bruce Langdon
LLNL
P.O. Box 808, L-472
Livermore, CA 94550

David Larson
LLNL
7000 East Avenue, MS L-630
Livermore CA 94550

William Lawson
NYU Courant Institute
251 Mercer Street
New York, NY 10012

W.W. Lee
PPPL
Theory Division
Princeton, NJ 08543

Paulett Liewer
JPL
MS 198-231
Pasadena, CA 91109

Sifeng Ma
UCLA
1-130 Knudsen Hall
405 Hilgard Avenue
Los Angeles, CA 90024

Janardhan Manickam
Princeton University
Box 451
Princeton, NJ 08543

Alan Mankofsky
SAIC
1710 Goodridge Drive
McLean, VA 22102

Yoshiyuki Matsuda
LLNL
P.O. Box 5511
Livermore, CA 94550

Michel McCoy
NERSC-LLNL
L-561
Box 5509
Livermore, CA 9455

Zoran Mikić SAIC 10260 Campus Point Drive, MS 32 San Diego, CA 92121-1578	John Petillo SAIC 1710 Goodridge Drive McLean, VA 22102	John Reynders PPPL Princeton University, Site C Princeton, NJ 08543
Al Mondelli SAIC 1710 Goodridge Drive McLean, VA 22102	Timothy Pointon SNL Division 1265, P.O. Box 5800 Albuquerque, NM 87185-5800	Helen Rudd RCS 6922 Nashville Road Lanham, MD 20706
Yuji Nakamura Kyoto Univ. Plasma Physics Laboratory Gokasho, Uji Kyoto 611 JAPAN	Sherrie Preische PPPL C-Site, Plasma Physics Lab. Princeton, NJ 08543	Robert Santoro PPPL Princeton University P.O. Box 451 Princeton, NJ 08543-0451
Richard Nebel LANL MS B217, T-15 Los Alamos, NM 87545	Richard Proccassini LLNL Inertial Confinement Fusion Program L-472, P.O. Box 808 Livermore, CA 94550	Florence Sausse CEA/CESTA Universite Bordeaux I BP No. 2 33114 Le Barp FRANCE
Cho-Kuen Ng SLAC P.O. Box 4349 Stanford University Stanford, CA 94309	Jeff Quintenz SNL Division 1265, P.O. Box 5800 Albuquerque, NM 87185-5800	Dalton Schnack SAIC 10260 Campus Point Drive, MS 32 San Diego, CA 92121-1578
William Nystrom LANL T-15, MS F647 Los Alamos, NM 87545	S. Rajagopalan SLAC Bin 26 Stanford, CA 94309	Petra Schutt TH Darmstadt Schloßgarten Str. 8 W-6100 Darmstadt GERMANY
Scott Parker PPPL Princeton University P.O. Box 451 Princeton, NJ 08543	Peter Rambo LLNL L-95 Livermore, CA 94550	David Seidel SNL Division 1265 P.O. Box 5800 Albuquerque, NM 87185
Michael Pasik SNL Division 1265, P.O. Box 5800 Albuquerque, NM 87185-5800	Carl Rathmann Cal Poly College of Engineering 3801 W. Temple Pomona, CA 91768-4066	Bill Sharp LLNL P.O. Box 808, L-626 Livermore, CA 94550

Magdi Shoucri
CCFM
1804 Montee Ste Julie
Varennnes, Quebec, J3X1S1
CANADA

Gregory Simonson
LLNL
L-84
Livermore, CA 94550

Steve Slinker
NRL
Code 4792
Washington, DC 20375-5000

Donald Stevens
New York Univ.
251 Mercer Street
New York, NY 10012

Owen Storey
NASA Goddard
NSSDC (Code 930.2)
Greenbelt, MD 20771

Motohiko Tanaka
Natl. Inst. Fusion Sci.
Ruroh-cho, Chikusa-Ku
Nagoya 464-01
JAPAN

Alfonso Tarditi
Univ. of California
Cory Hall
EECS Dept.
Berkeley, CA 94720

SW Temko
Moscow Geological Prospecting Institute
MGPI, Mikhlucho-Maklaia Street 23
117135 Moscow, B-485
USSR

Vahid Vahedi
Univ. of California
207-209 Cory Hall, ELCS Dept.
Berkeley, CA 94720

Erik Vold
LLNL
MS-F664
Los Alamos, NM 87545

Scott von Laven
MRC
1720 Randolph Road
Albuquerque, NM 87106-4245

Hoanh Vu
LANL
B 216, Group T-3
Los Alamos, NM 87545

Thomas Westermann
FK Karlsruhe
Postfach 3640
Karlsruhe 7500
GERMANY

Timothy Williams
LLNL
P.O. Box 5511, L-630
Livermore, CA 94550

Peter Winkler
Univ. of Nevada
Department of Physics/220
Reno, NV 89557-0058

Earnest Zaidman
NRL
Code 6840
Washington, DC 20375-5000

Richard Zwakenberg
NERSC
P.O. Box 5509
Livermore, CA 94550

Author Index

- Adolf, A -- OW3
- Ambrosiano, J. -- OW2
- Anderson, D.V. -- PWE14
- Anker, W. -- PWE5, PWE15
- Assous, F. -- OW3
- Barnard, J.J. -- PWA3
- Barnes, D.C. -- PWA15, PWE6
- Bateman, G. -- PWA12
- Bertrand, P. -- PT2, PT14, PT15
- Bezmerov, I.V. -- PWE8
- Blank, A.A -- PT10
- Bouquet, S. -- OT7
- Brackbill, J.U. -- PWA13, PWE18, PT9, OF10
- Brandon, S.T. -- OW2
- Byers, J.A. -- PT6
- Callahan, D.A -- PWE7
- Campbell, P.M. -- PT1
- Caramana, E.J. -- OT6
- Cargill, P.J. -- PWE9
- Chen, P. -- PWA9, PT18
- Chernin, D.P. -- PWE19
- Church, B.W. -- PWE10
- Coats, R.S. -- PWE1, PT11
- Dannevik, W.P. -- PWA18
- Daverton, F. -- PT13
- Dawson, J.K. -- PWA7, OT1
- Decyk, V.K. -- OT1, OT3
- DeFainchtein, R. -- PT19
- Degond, P. -- OW1, OW3
- Delaurens, F. -- PT13
- Dey, A.K. -- OF1
- Dimitis, A.M. -- PWA5
- DiPeso, G. -- PWA11
- Dorland, W. -- PWA6
- Drobot, A.T. -- OW4, OW7, PWA14, OF6
- Eastwood, J.W. -- PWE15
- Eideman, S. -- OW7
- Eppley, K. -- PWA1
- Esarey, E. -- PT4
- Feix, M.R. -- OT7, PT2, PT14, PT15, OF2
- Ferraro, R.D. -- OT1, OT3
- Fijalkow, E. -- PT2, PT14, PT15
- Freund, H.P. -- PT16
- Friedman, A. -- PWE7, PWE16, CF5
- Furnish, G. -- OT4, PT3
- Ganguly, K. -- OF1
- Geary, J.L. -- PWA17
- Ghizzo, A. -- PT2, PT14, PT15
- Ghosh, S. -- PT19
- Gill, L. -- PT3
- Glasser, A.H. -- OW6
- Goldstein, M.L. -- PT19
- Goodrich, C.C. -- PWE9
- Gray, M.G. -- OT4
- Gritz, L.A. -- OF1
- Grote, D.P. -- PWE7, PWE16, OF5
- Haber, I. -- PWE3, PWE7, PWE16, OF5
- Hammett, G.W. -- PWA6
- Harold, J.B. -- PT17
- Hassam, A.B. -- PT17
- Harvey, R.W. -- PWE20
- Heintz, E. -- OW3
- Hermeline, F. -- OW3, PT13
- Hewett, D.W. -- PT6, PT7
- Hirshman, S.P. -- PWE11, PWE13
- Holdaway, R. -- PWA17
- Hubbard, R.F. -- PWA4
- Ichiguchi, K. -- PWE13
- Jamin, E. -- PT2
- Johnson, J.L. -- PWE2, PWE11, PWE13
- Johnston, T. -- PT14

Jones, M.E -- OF8	Matsuda, Y. -- PT5	Rudd, H. -- PWE3
Joyce, G -- FWA2, PWA4, PT4	McCoy, M.G. -- PWE20	Salas, A. -- PWE11
Karttunen, S. -- PT14	Mikic, Z. -- OW7	Salmaa, R. -- PT14
Kerbel, G.D -- PWE20	Mondelli, A.A. -- PWE19, OF6	Santoro, R.A. -- PWA8
Kieler, M.L -- PWE1, PT11	Monicello, D.A. -- PWE11	Sato, T. -- OF7
Kostas, C -- PWE19	Murakami, S. -- OF7	Saussede, F. -- PT8
Kotschenreuther, M. -- PT20	Nakamura, Y. -- PWE13	Schnack, D.D. -- OW7
Krall, J -- PWA2, PWA4, PT4	Nebel, R.A. -- PWE6	Schuett, P. -- OF4
Kress, M.E -- PT1C, OF6	Ng, C.-K. -- PWA9, PT18	Segre, J. -- OW3
Krücken, T -- OT1	Nystrom, W.D. -- PWA15, OT4	Seidel, D.B. -- PWE1, PT11
Krueger, W.A. -- PWA14, OF6	Oliver, B.V. -- PWE10	Sharp, W.M. -- PWA3
Kuz'min, S.K. -- PWA10	Parker, S.E. -- OF9	Shen, D. -- OF1
Kuzmin, S.V. -- PWE17	Pasik, M.F. -- PT11	Shoucri, M. -- PT2, PT14, PT15
Langdon, A.B -- PWE7, PWE16, OF3	Pattikangas, T. -- PT14	Silakov, V.P. -- PWE8
Larson, D.J. -- PT7	Patillo, J.J. -- PWE19, OF6	Simnson, G.F. -- PT6
Lawson, W.S. -- PWE12	Pointon, T.D. -- PWE1, PT11	Siregar, E. -- PT19
LeBrun, M.J. -- PWA15, OT4, PT3	Preisiche, S. -- PWE2	Slinker, S. -- PWA2, PWA4
Lee, W.W. -- PWA8, OT5, OF9	Procassini, R.J. -- PWA18	Smith, R.A. -- PWA17
Liewer, P.C. -- OT1, OT3	Quintenz, J.P. -- PWE1, PT11	Strauss, H.R. -- PWE12
Löhner, R. -- OW2	Rajagopalan, S. -- PWA9, PT18	Sudan, R.N. -- PWE10
Longcope, D.W. -- PWE10	Rambo, P.W. -- PT12	Surendra, M. -- PWA11
Lottati, I. -- OW7	Raviart, P.A. -- OW3	Sydora, R.D. -- PWA7
Ma, S. -- PWA7	Reiman, A.H. -- PWE11	Tajima, T.T. -- PWA15, OT4
Manfredi, G. -- OT7	Rewoldt, G. -- PWE13	Takamaru, H. -- OF7
Manickam, J. -- PWE2	Reynders, J.V.W. -- OT5	Tanaka, M. -- OF7
Manikofsky, A. -- OF6	Roy, S.S. -- OT5	Temko, K.W. -- PWA10

Temko, S.W. -- PWA10
Thomas, V.A. -- OF8
Tsang, K.T. -- PWE19
Vahedi, V. -- PWA11, PWE4
Verboncoeur, J.P. -- PWA11, PWE4
Vold, E.L. -- PWA16
von Laven, S.A. -- PT1
Vu, H.X. -- PWE18, PT9
Wakatani, M. -- PWE13
Weiland, T. -- OF4
Westermann, T. -- OW5
Whitman, S.R. -- PWA18
Winske, D. -- OF8
Williams, T.J. -- PT5
Yu, S.S. -- PWA3
Zaidman, E. -- OT2
Zalesak, S. -- PT19